Apelon, Inc.
Suite 202, 100 Danbury Road
Ridgefield, CT 06877

Phone: (203) 431-2530
Fax: (203) 431-2523
www.apelon.com

Apelon Distributed Terminology System (DTS)

Subscription Content Import Guide

Table of Contents

Introduction

After you **create** the DTS schema, one of your options for populating the generated tables with data is to perform a **Full Import** of initial subscription data. When you select this option, Apelon forwards to you, in a **.zip** file, all initial subscription data from the vocabularies (also referred to as **namespaces**) to which you subscribe.

You then can use the **DTS Subscription Import Wizard** to import the initial namespace subscription data into your knowledgebase tables. You also can use the Subscription Import Wizard to import namespace **content updates** forwarded to you from Apelon based on your subscription agreements.

For a Linux installation, you can execute **subscriptionImportWizard.sh** in the **bin** subdirectory to start the Subscription Import Wizard.

For a *Windows* installation, you also can perform full and update content imports by running the **kbcontent-import.bat** file (**bin\kb\content\import**).

For a Linux installation, you can execute **kbcontent-import.sh** in **bin/kb/content/import** to perform full and update content imports.

Procedures for performing full and update imports using both of these methods are included in this guide.

Audience for This Guide

The *Subscription Content Import Guide* was written as a reference for personnel who perform the imports of subscription namespace content (forwarded from Apelon) into DTS. The imported content can be an entire namespace (as in an initial, **Full** import) or an import of namespace updates only.

Populate New Tables Through "Full" Subscription Import

After you run the **Knowledgebase Create** utility to create the DTS schema (refer to the *Create DTS Schema* discussion in the *Knowledgebase Administrators Guide*), you must populate the DTS tables with either content data migrated from another database (e.g., from TDE), or content data from a subscription namespace. If you intend to populate the tables with **migrated content**, you must run the **Knowledgebase Load** utility (refer to the *Populate DTS Tables Through Data Migration* discussions in the *Knowledgebase Administrators Guide* for migration procedures).

If you intend to populate the tables with content from a **subscription** namespace, you must perform a **full import** of the initial subscription data into DTS. Apelon exports initial subscription data to you in a **.zip** file that includes data from all of the vocabularies (referred to in DTS as **namespaces**) to which you subscribe.

Perform "Full" Import Using the DTS Subscription Import Wizard

DTS provides a GUI-based subscription import tool called the **DTS Subscription Import Wizard*** . This wizard guides you through the import of initial, baseline subscription content for each of your subscription namespaces.

You subsequently can use the Import Wizard to import **updates** to your subscription namespace content. These scheduled updates are provided to you by Apelon based on your subscription agreements.

Follow this procedure to use the Import Wizard to perform a **full** import of initial subscription namespace content.

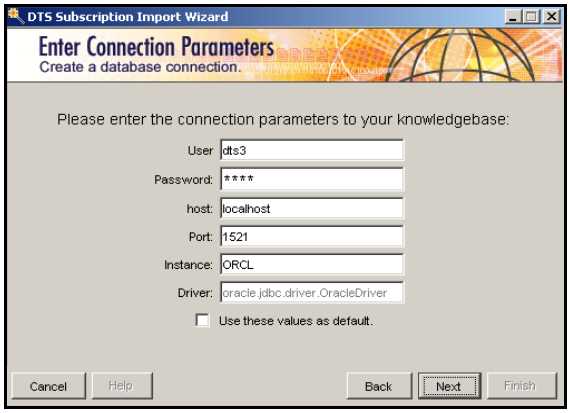
1. Download the **.zip** import subscription data file from Apelon into a directory on your machine.
2. Select **Subscription Import Wizard** from the *Windows* **Start** menu (**Apelon**> ***DTSInstall ,*** where *DTSInstall* is the selected installation directory>**Subscription Import Wizard**).

   > For a Linux installation, execute **subscriptionImportWizard.sh** in the **bin** subdirectory to start the Subscription Import Wizard.

   The *Import Wizard Welcome* window displays.

3. Ensure that the database type selected (**Oracle**, the default, or **SQL Server**) reflects your DTS Knowledgebase type.

4. Click **Next**. The *Connection Parameters* window displays.

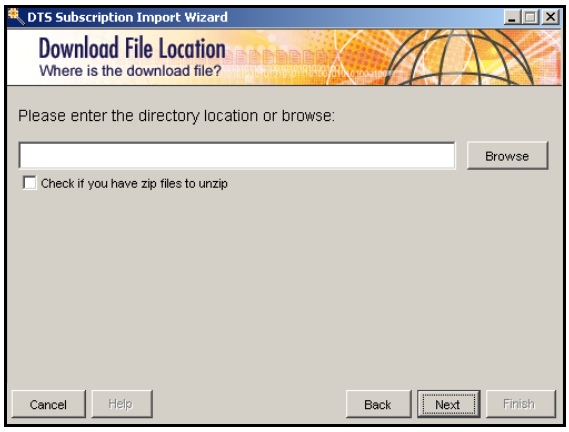


   If you need to modify the database type for your DTS Knowledgebase, click **Back** to return to the *Welcome* window.

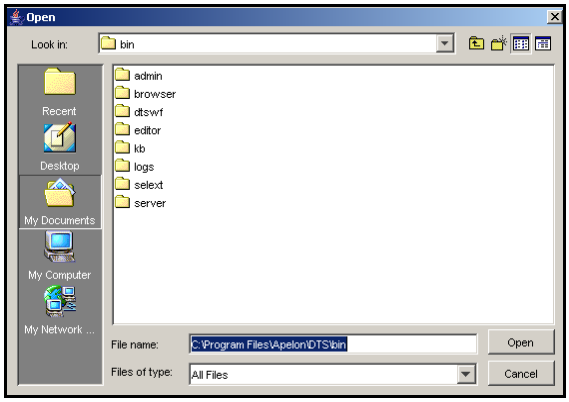
   On the *Connection Parameters* window you specify the location of your DTS Knowledgebase, and the parameters for connecting to it for the purpose of importing new subscription content. The default connection parameters are from the **kbimportwizard.xml** file (***DTSInstall* \bin\kb**).

   You can modify the default connection parameters as necessary. Click **Use these values as default** to retain these parameters for future sessions.

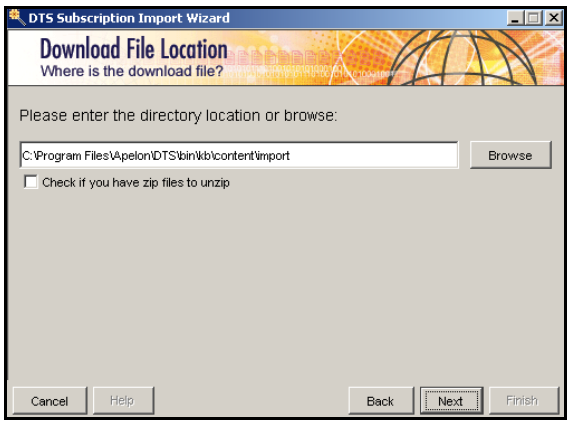
5. Click **Next**. The *Download File Location* window displays. Here you specify the directory where you want to extract (i.e., unzip) the import **.zip** file from Apelon. This directory can differ from the one where you downloaded the **.zip** file.



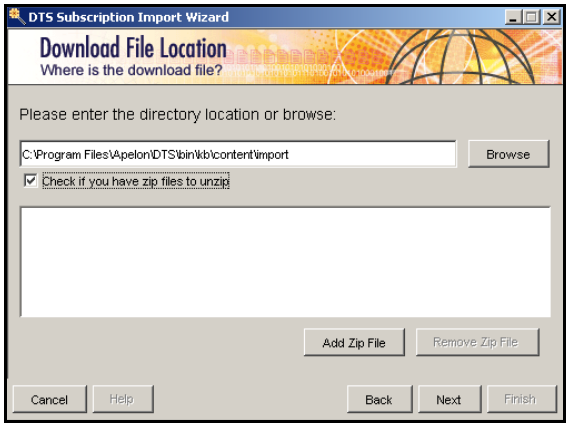
Click the **Browse** button to display the *Browse* window.



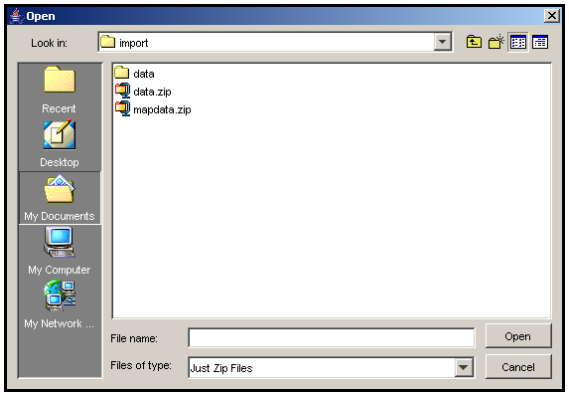
6. Browse to the directory where you want to unzip the downloaded **.zip** subscription import file (or where the downloaded file was unzipped manually) then click **Open**. The *Download File Location* window redisplays, referencing the directory path you selected.



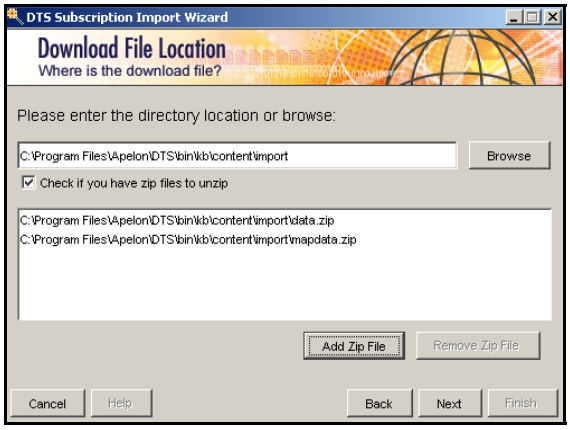
7. If you have not unzipped the **.zip** subscription import file you downloaded from Apelon, click the option box for unzipping the file (this inserts a check mark). A view area displays in the lower portion of the *Download File Location* window.



8. Click **Add Zip File**. This indicates that you want to select, and unzip, the **.zip** file that contains your subscription import information. Another *Browse* window displays.



Browse to the directory where the **.zip** subscription import file is located, click the **.zip** file name to highlight it, then click **Open**. The directory path and **.zip** file name display in the view area on the *Download File Location* window. Note that you select each **.zip** file individually; repeat this step to select multiple files.



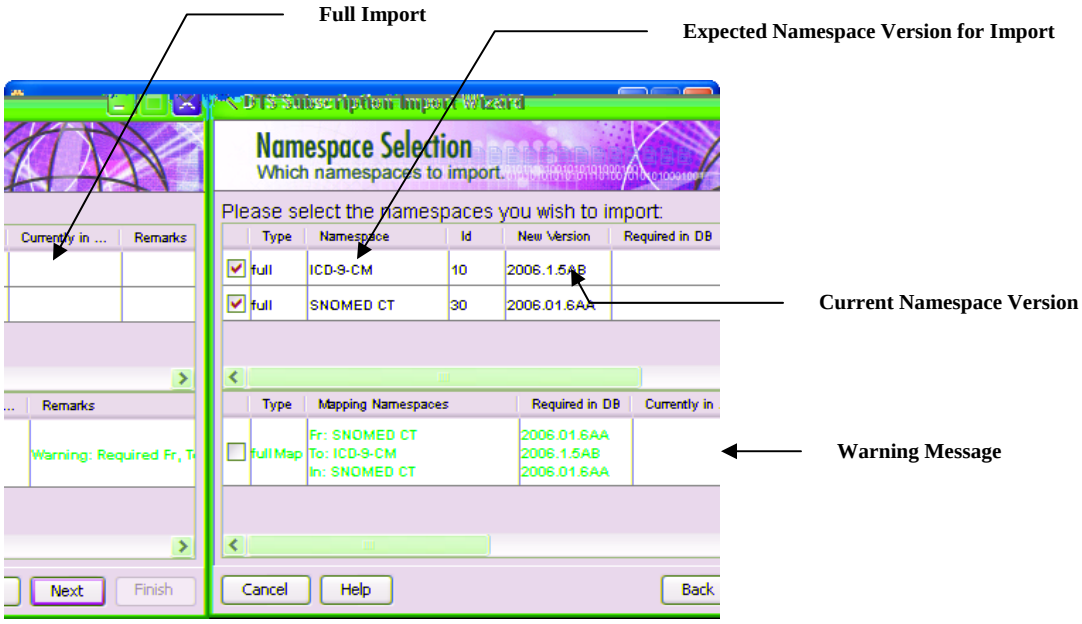
9. Click **Next**. A window displays indicating that import files are being unzipped.

As part of the process a **data** subdirectory is created in that directory. Under the **data** directory, a separate subdirectory is created representing each namespace for which you can import content.

When the unzip process completes, the *Namespace Selection* window displays. This window lists each namespace content file (and any associated mapping file) that was included in the import **.zip** files you downloaded. The **Type** column indicates **full** for each namespace for which you are performing a full import, or **fullMap** for each full mapping import.

The latest (expected) namespace version that is required to be present in the target database is indicated under **Required in DB**. The current namespace version currently in the target database is indicated under **Currently in DB**.

In this illustration, the **Currently in DB** column does not reference a previous database (because the import is an **initial** FULL import of subscription content).



10. The default is to import all listed content and mapping files from the .zip file. To deselect an individual file for the import, click the corresponding checkbox in the column adjacent to the **Type** column to remove the check mark.

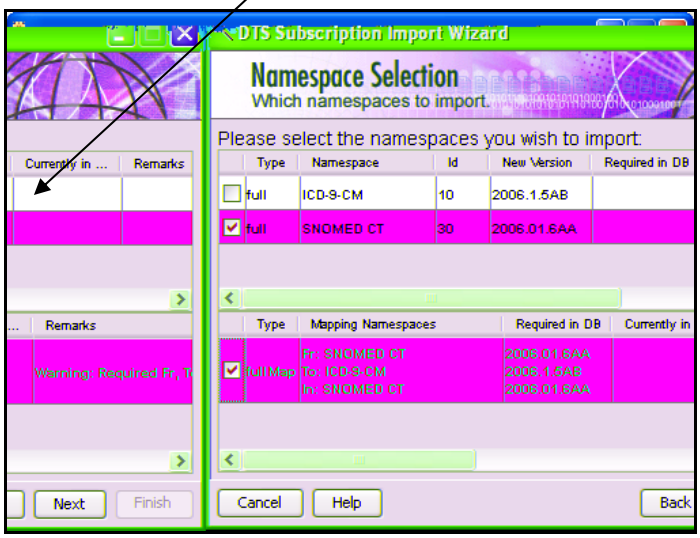
    In the **Mapping Namespace** section of the window illustrated previously, the following message displays under **Remarks**, and the line is highlighted in blue text, to indicate a **Warning** scenario.

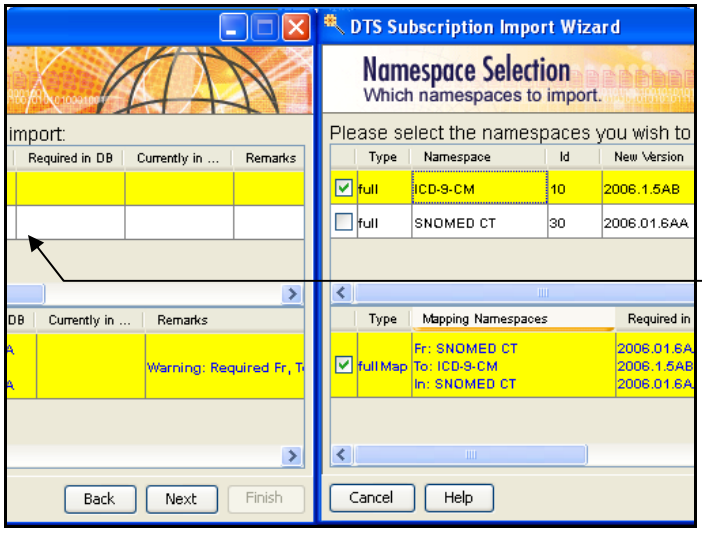
    Warning: Required Fr, To or In namespaces not currently present in knowledgebase. Ensure that required namespaces are selected as part of import if not already present.

    In this case, successful import of the **Mapping** namespace is contingent on the successful **Full** import of the two selected namespaces (with the required versions). Since **FULL** import of the two namespaces will occur first, you can check the **fullMap** box to start the import.

In each of the following illustrations, one of the FULL import namespaces is not checked for import.

Namespace Required for Mapping Import is Unchecked





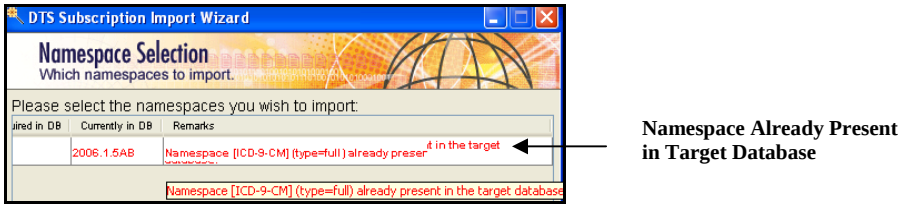
Namespace Required for Mapping Import is Unchecked

Even though you are permitted to select the **fullMap** mapping namespace for import, that mapping import will fail unless you first select the required namespace(s) for **FULL** import.

11. The subscription import content for each namespace is accompanied by a configuration file (**kbcontent-import-full.xml** for content, **kbmap-import-full-[source_nsp_id]-[target_nsp_id].xml** for mapping) that includes information regarding the version of the current target namespace (if any) into which you are importing new content. Using this information, the Import process will perform an integrity check to verify that the current (target) namespace version is compatible with the version you want to import. For a **full** import, the check verifies that no current version is present in the target database (as illustrated).

The import of a specific namespace will be prevented if the process detects a version incompatibility between import namespace content and the content in the target database. The incompatibility is indicated in the *Namespace Selection* window if the values in the **Required in DB** and the **Currently in DB** fields are different. The checkbox for the namespace that is adjacent to the **Type** column is disabled, the text on that line is highlighted in red, and the **Remarks** column indicates the situation preventing the import.

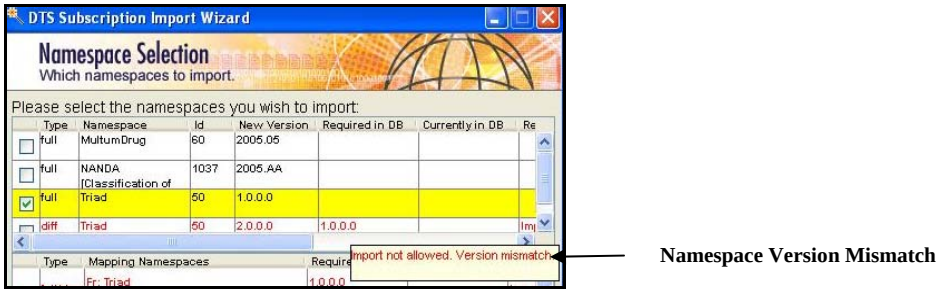
In the following illustration, the subscription content version being imported exists already in the target namespace.



Namespace Already Present in Target Database

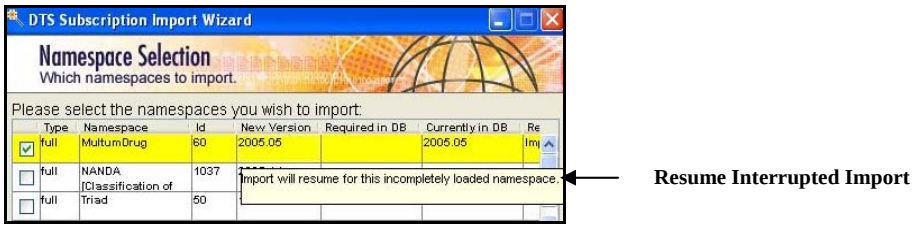
Note that the tooltip text box also indicates the incompatibility issue.

In the next illustration, a version mismatch is indicated (i.e., the namespace version under **New Version** differs from the version under **Required in DB**). This situation can occur if successive subscription updates are missed by a client.



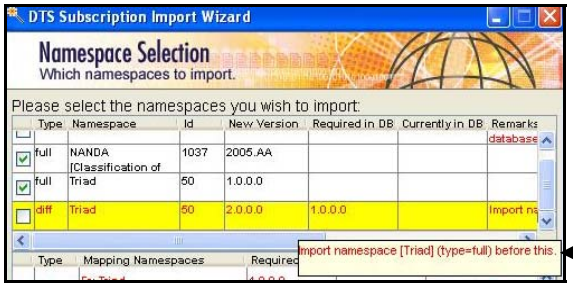
Namespace Version Mismatch

In the next illustration, an import that was interrupted earlier is being resumed (the namespace checkbox adjacent to the **Type** column is enabled in this case).



Resume Interrupted Import

The next illustration indicates that an update (i.e., **diff**) is being disabled because a **full** import of the namespace must be performed first.

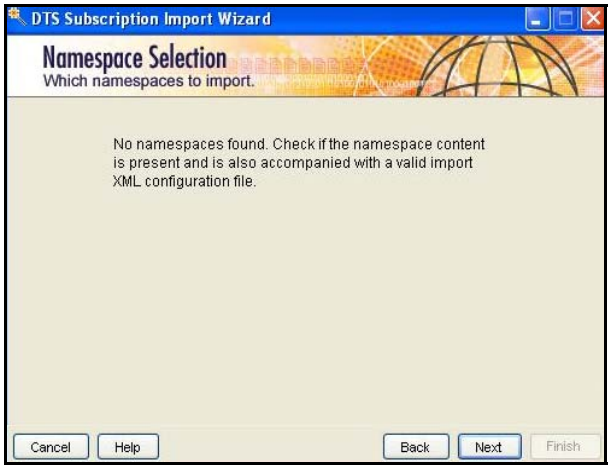


Full Import Required Prior to Update

The Import process will prevent import of content for a specific namespace until you reconcile the version incompatibility referenced for that namespace.

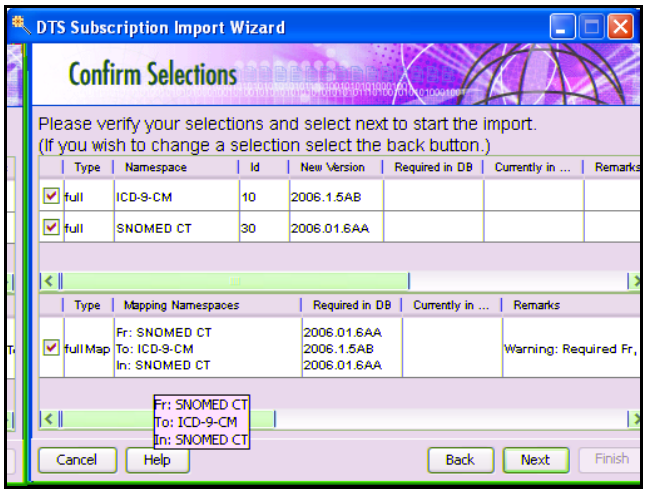
Import Content From DTS Versions Prior to 3.4

You cannot import content from DTS Version 3.3 (and older) unless there is a valid Import Integrity Verification configuration file (**kbcontent-import-full.xml** for content, **kbmap-import-full-[source_nsp_id]-[target_nsp_id].xml** for mapping) included with that older subscription content. If the configuration file is not included, the following window displays.



The Import Integrity Verification file must be configured manually, then added to this older content before that content can be imported into DTS. Refer to the *Generate Import Integrity Verification File* discussion in the *Knowledgebase Administrators Guide* for instructions on generating this configuration file, and including it with your older subscription content for import into DTS.

12. When you complete namespace and mapping file import selections, click **Next** (you must make at least one namespace or mapping import selection in order to advance). The *Confirm Selections* window displays for you to verify your selections.



If you need to modify your selections, click **Back**. When the *Namespace Selection* window redisplays you can modify your selections, as needed.

13. You have the option to define the order in which namespaces will be imported. Open the **import-precedence.txt** file in ***DTSInstall*** **\bin** (the file is illustrated).

```
# Format for importing namespace
#
# <namespaceId>|<importType>
# Legal values for namespace importType: full or diff
# Example:
# 10|full
# 30|full
# 10|diff
#
# Format for importing mappings
#
# <fromNamespaceId>-<toNamespaceId>|<importType>
# Legal values for mapping importType: fullMap or diffMap
# Example:
#
# 30-10|fullMap
# 30-35|fullMap
# 30-10|diffMap
# 30-35|diffMap


# Import Options

# When set to yes, the import will not disable table constraints
# and drop indexes. By default it is set to no.
skipConstraints=no

# When set to yes, the import will run analyze tables (oracle only)
# By default it is set to yes.
runPostLoadProcesses=yes

# Pre-processing tasks
# Provide a file name that has tasks listed which are required to be run
# before the import process
#
# pre=

# Post-processing tasks
# Provide a file name that has tasks listed which are required to be run
# after the import process
#
# post=
```

List of Namespace IDs

Modify the list of source namespace ID numbers to reflect the order in which you want to import the namespaces. Note that namespaces always are imported prior to their mappings to preserve database referential integrity.

Also included in the **import-precedence.txt** file are the **-pre** and **-post** switches. You use these switches to define tasks that can be run prior to the import process, or after the import completes (e.g., addition of indexes to archive tables **prior** to the import, deletion of archive table indexes **after** import completion). You can use this feature to customize the import process with respect to performance improvements and tuning; Apelon subscriptions may include these options depending upon the size of the content updates.

Each switch must be accompanied by the task file name for the task to be run. The sample task file **pre-task-list-sample.txt** (in ***DTSInstall*** **\bin**) defines a task for the addition of indexes prior to the import run. The sample task file is illustrated.

```
# Processes List Sample
#
# Format :
# executable=<executable-name>;workingDir=<working directory path (absolute)>
#
# Example:
# executable=add-my-indexes.bat;workingDir=D:\Program Files\Apelon\DTS\scripts\Oracle
```

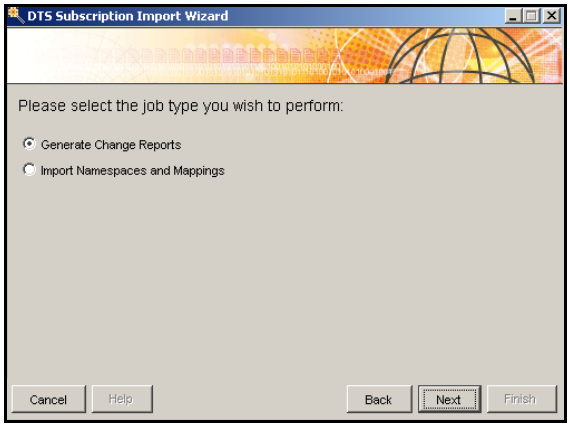
The required entry in the **import-precedence.txt** file is illustrated.

```
# Pre-processing tasks
# Provide a file name that has tasks listed which are required to be run
# before the import process
#
# pre=pre-task-list-sample.txt

# Post-processing tasks
# Provide a file name that has tasks listed which are required to be run
# after the import process
#
# post=
```

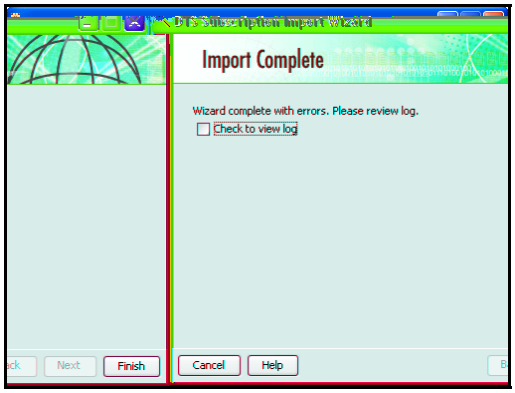
Task File Name

14. Click **Next**. The following window displays.



15. Click *Import Namespaces and Mappings* to begin the import process. A status window displays to indicate the progress of the import. A confirmation window displays when the import is completed.

If errors occurred during the import, click the *View Log* option to review them.



16. Click **Finish** to exit the Subscription Import Wizard.

If you cancel an import that is in progress by clicking the **Cancel** button, you can resume the import at the point at which it was stopped. Click *Import Namespaces and Mappings* to restart the import process at the point where it was interrupted.

***** A modified version of the open source Java Wizard Framework was used in development of the DTS Subscription Import Wizard, which is included in this release. The Java Wizard framework and its use are covered by the GNU Lesser General Public License. A copy of the license is included in the **license.txt** file in the following installed DTS subdirectory:

***DTSInstall* \samples\wizard\license.txt**

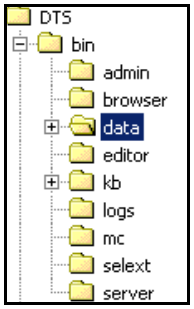
You also can click the following link to view the license:

http://www.opensource.org/licenses/lgpl-license.php.

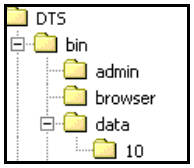
Perform "Full" Import Using kbcontent-import.bat or kbcontent-import.sh

Follow this procedure to perform a **full** subscription namespace import into your DTS tables by running **kbcontent-import.bat** (for a DTS *Windows* installation) or, for a DTS Linux installation, by executing **kbcontent-import.sh** in (**bin/kb/content/import**).

1. Unzip the **SOURCE-VERSION-FULL.zip** import subscription data file provided by Apelon into the ***DTSInstall* \bin** directory. A **data** subdirectory is created (if not present already) in ***DTSInstall* \bin**.



Under the new **data** directory, a separate subdirectory is created for each terminology source (i.e., namespace) in your knowledgebase. Note the illustration.



2. At this point you must edit the **kbcontent-import.bat** file in ***DTSInstall* \bin\kb\ content\import** for the import. (For Linux installations, refer to the discussion on executing **kbcontent-import.sh** later in this section.)

   As of DTS Version 3.4, the subscription import content for each namespace is accompanied by a configuration file (**kbcontent-import-full.xml** for content, **kbmap-import-full-[source_nsp_id]-[target_nsp_id].xml** for mapping). Each file includes information regarding the version of the current target namespace (if any) into which you are importing new content. The configuration file for each namespace to be imported will be in ***DTSInstall* \bin\data.**

   Using this information, the Import process will perform an integrity check to verify that the current (target) namespace version is compatible with the version you want to import. The import of a specific namespace will be prevented if the process detects a version incompatibility between import namespace content and the content in the target database.

   Note the highlighted switches in the illustrated **kbcontent-import.bat** file.

```
@echo off
REM
REM Process Arguments : -p <properties-file.xml> -c <configuration full-class name>
REM

REM
REM Process arguments overriding the properties file configuration variables:
REM (i.e., connection params from a separate file overrides one in the property file):
REM Usage: -c <system configuration class name> which is mandatory
REM         -action [export, import, export_mappings, import_mappings, content_change_report, publish_namespace]
REM                 export -> exporting content from DTS
REM                 import -> importing content into DTS
REM                 export_mappings -> exporting inter-source mappings (associations) from DTS knowledgebase
REM                 import_mappings -> importing inter-source mappings (associations) into DTS knowledgebase
REM                 content_change_report -> to generate reports showing concept associations affected by subscription
content update
REM                 publish_namespace -> publish a local namespace
REM         -p <user property file name>
REM                 XML file having the properties needed to run this process (supplied with namespace content).
REM         -skipConstraints (yes/no)
REM                 If yes, the process does not disable/enable constraints and drop/add indexes while updating DTS
knowledgebase
REM         -t <target connection config file>
REM                 file with the target connection params.
REM                 The application loads connection params from this file, and NOT from kbcontent-import-
{full/diff}.xml
REM         -d <data extract directory>
REM                 path to folder where subscription content is to be read from/written to.
REM         -runPostLoadProcesses <post load processing (yes/no) {optional}, default=yes>
REM          [e.g. analyze tables in case of Oracle]
REM       -pre  <pre processes file name>
REM       -post <post processes file name>
REM
REM -- Imports a given namespace (full)

cd ..\..\..

REM
REM Switch -t kb/target-connection.xml is required here as the connection parameters are read from the target-
connection.xml file.
REM                                                    Modify -t kb/target-connection.xml Switch

call runApp_cw 512 com.apelon.dts.db.admin.DbContentMgr -c com.apelon.dts.db.admin.config.DBContentMgrConfig -p
data/{namespace_id}/kbcontent-import-{full/diff}.xml -action import -t kb/target-connection.xml
cd kb\content\import
```

Modify -p data/{namespace_id}/kbcontent-import-{full/diff}.xml Switch

Edit the **-p data/{namespace_id}/kbcontent-import-{full/diff}.xml** switch to reference the import configuration file in ***DTSInstall* \bin\data\ [namespace_id]** that accompanied the namespace content; include a switch for each source namespace.

You can edit the **kbcontent-import.bat** file (or **kbcontent-import.sh,** for Linux) to reference a custom connection configuration file that will not be overwritten during the subscription import process. The template connection file **target-connection.xml** allows you to define a custom connection based on the **target**

connection parameters for content import.
To establish a target connection in the **target-connection.xml** file, the value of the **direction** property should be **target**. Note the illustration.

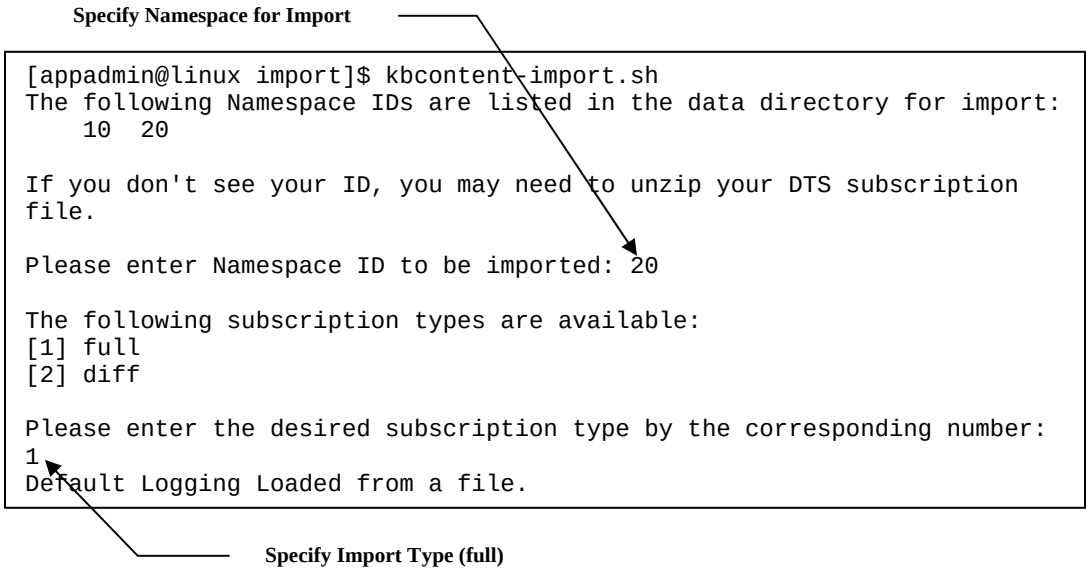
<property name="direction" value="**target**" />

You can create multiple versions of the **target-connection.xml** file, each with its own specific name, and each reflecting a specific connection from, or to, a data source.

Modify the **-t kb/target-connection.xml** switch to reference the appropriate target-connection.xml file. Apelon recommends that you rename each modified kbcontent-import.bat batch file (or **kbcontent-import.sh,** for Linux) to reflect its purpose (e.g., **kbcontent-import-test.bat**, **kbcontent-import-production.bat**, etc.).

3. Run **kbcontent-import.bat** (or **kbcontent-import.sh,** for Linux) to import initial subscription content into your knowledgebase. Content from each namespace source referenced in the **kbcontent-import.bat** file (or **kbcontent-import.sh**) is imported from its corresponding subscription namespace subdirectory under ***DTSInstall* \bin\data**.

   When you execute **kbcontent-import.sh** (**bin/kb/content/import**) for Linux, you are prompted to specify the IDs for the namespaces you want to import, and indicate if this is a **full** (**1**) or **diff** (**2**) import.



   If the Import process detects a version incompatibility that would cause the individual namespace import to fail, the import of that namespace is prevented.

You must reconcile the version incompatibility for the namespace that failed the import, then attempt the import again.

Import Content From DTS Versions Prior to 3.4

You cannot import content from DTS Version 3.3 (and older) unless there is a valid Import Integrity Verification configuration file (**kbcontent-import-full.xml** for content, **kbmap-import-full-[source_nsp_id]-[target_nsp_id].xml** for mapping) included with that older subscription content.

This Import Integrity Verification file must be generated and added to this older content before you can import that content into DTS. Refer to the *Generate Import Integrity Verification File* discussion in the *Knowledgebase Administrators Guide* for instructions on generating this file, and for including it with your older subscription content for import into DTS.

Import Namespace Subscription Updates

Perform "Diff" Import Using the DTS Subscription Import Wizard

You can use the **DTS Subscription Import Wizard*** to import namespace updates (called **diff** content imports) forwarded to you from Apelon. These scheduled updates are provided to you by Apelon based on your subscription agreements. Apelon **strongly** recommends that you perform a database backup prior to each subscription update import.

DTS allows you to create new content for each Ontylog subscription namespace in a linked **Extension** namespace. Each Extension namespace extends the content in a specific linked subscription namespace. Ontylog namespace concepts are organized into a hierarchy by a process called **classification**, a process you can perform using the DTS Editor.

In order to run classification for an Extension namespace, the linked subscription namespace must have a Classification Graph established for it. Refer to the *Import Ontylog Subscription Updates With "Classification" Graph* discussions in the *Ontylog Extension Namespaces and Extension Namespace Classification in DTS* document for more on classification graphs.

> Note that using the **Knowledgebase Publishing** utility provided with Apelon DTS, you also can publish your local namespace content on a regular basis, and forward your own updated content to **your** subscribers. Refer to the *Publish Client's Local Namespace* discussions in the *Knowledgebase Administrators Guide* for procedures.

Follow this procedure to use the Import Wizard to perform a **diff** import of updated subscription namespace content.

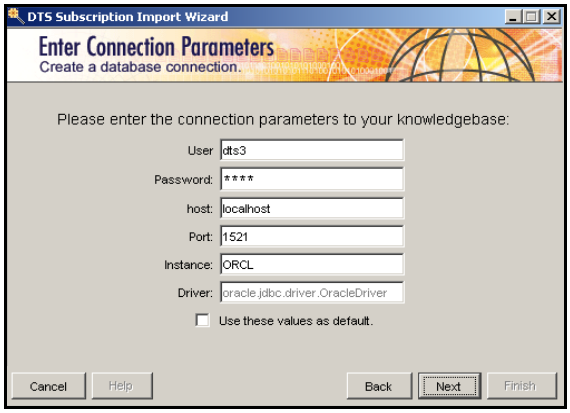
1. Download the **.zip** import subscription data file from Apelon into a directory on your machine.
2. Select **Subscription Import Wizard** from the *Windows* **Start** menu (**Apelon> *DTSInstall* >Subscription Import Wizard**).

   > For a Linux installation, execute **subscriptionImportWizard.sh** in the **bin** subdirectory to start the Subscription Import Wizard.

   The *Subscription Import Welcome* window displays.

3. Ensure that the database type selected (**Oracle**, the default. or **SQL Server**) reflects your DTS Knowledgebase type.

4. Click **Next**. The *Connection Parameters* window displays.

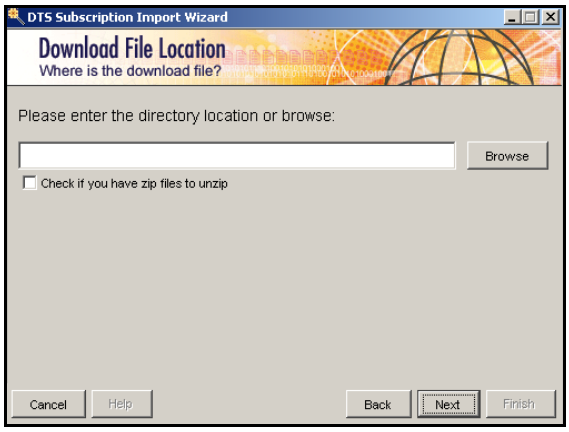


   If you need to modify the database type for your DTS Knowledgebase, click **Back** to return to the *Welcome* window.

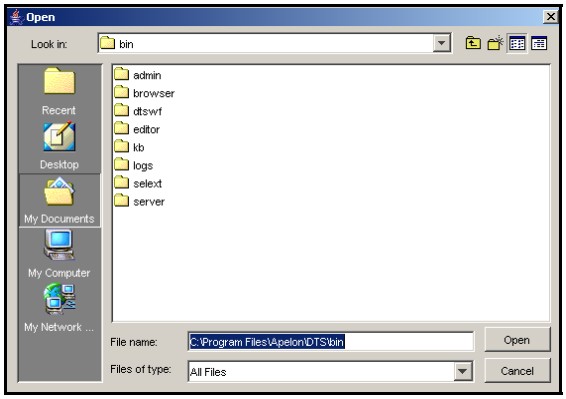
   On the *Connection Parameters* window you specify the location of your DTS Knowledgebase, and the parameters for connecting to it for the purpose of importing subscription content updates. The default connection parameters are from the **kbimportwizard.xml** file (***DTSInstall*\bin\kb**).

   You can modify the default connection parameters as necessary. Click **Use these values as default** to retain these parameters for future sessions.

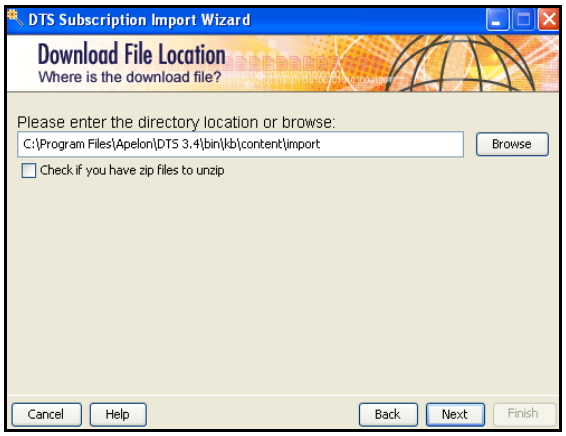
5. Click **Next**. The *Download File Location* window displays. Here you specify the directory where you want to extract (i.e., unzip) the import **.zip** file from Apelon. This directory can differ from the one where you downloaded the **.zip** file.



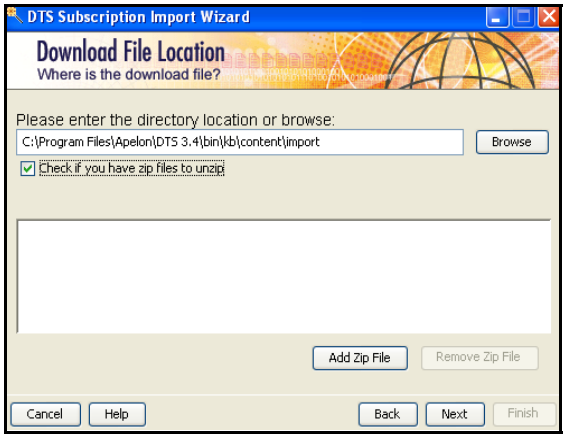
Click the **Browse** button to display the *Browse* window.



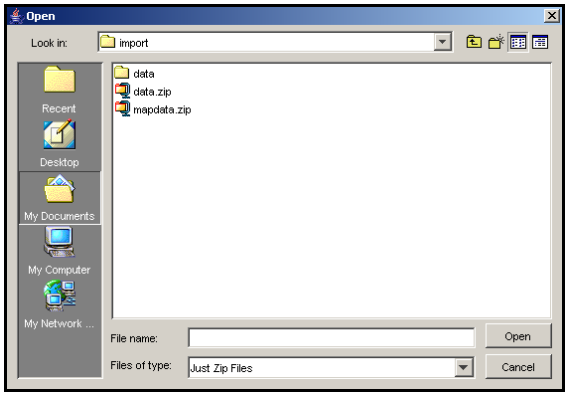
6. Browse to the directory where you want to unzip the downloaded **.zip** subscription import file (or where the downloaded file was unzipped manually) then click **Open**. The *Download File Location* window redisplays, referencing the directory path you selected.



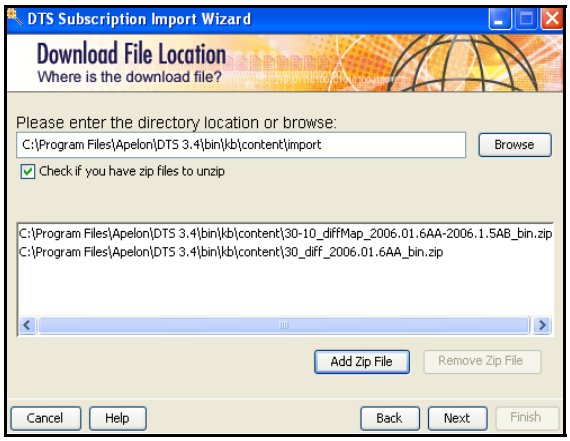
7. If you have not unzipped the **.zip** subscription import file you downloaded from Apelon, click the option box for unzipping the file (this inserts a check mark). A view area displays in the lower portion of the *Download File Location* window.



8. Click **Add Zip File**. This indicates that you want to select, and unzip, the **.zip** file that contains your subscription import information. Another *Browse* window displays.



Browse to the directory where the **.zip** subscription import file is located, click the **.zip** file name to highlight it, then click **Open**. The directory path and **.zip** file name display in the view area on the *Download File Location* window. Note that you select each **.zip** file individually; repeat this step to select multiple files.

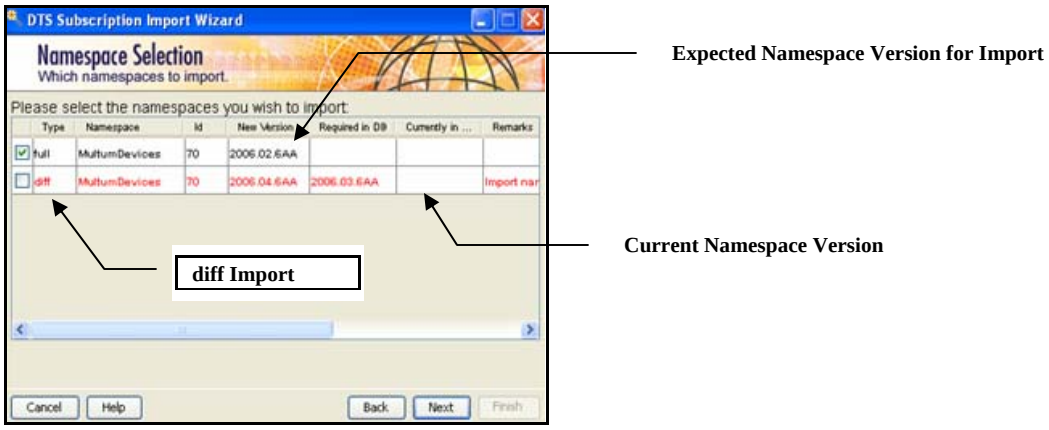


Click **Next**. A window displays indicating that import files are being unzipped.

As part of the process a **data** subdirectory is created in that directory. Under the **data** directory, a separate subdirectory is created representing each namespace for which you can import content.

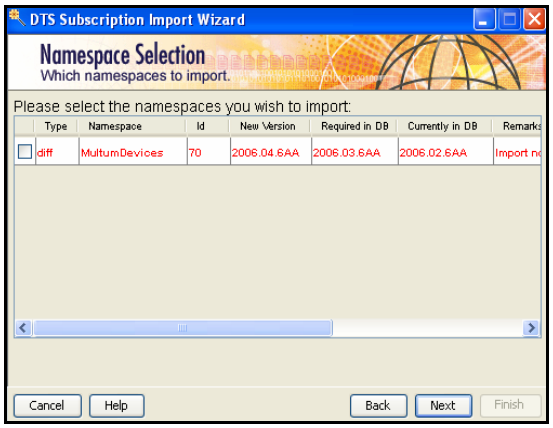
When the unzip process completes, the *Namespace Selection* window displays. This window lists each namespace content file (and any associated mapping file) that was included in the import **.zip** files you downloaded. The **Type** column indicates **diff** for each namespace for which you are performing a diff import, or **diffMap** for each diff mapping import.

The latest (expected) namespace version that is required to be present in the target database is indicated under **Required in DB**. The current namespace version currently in the target database is indicated under **Currently in DB**.

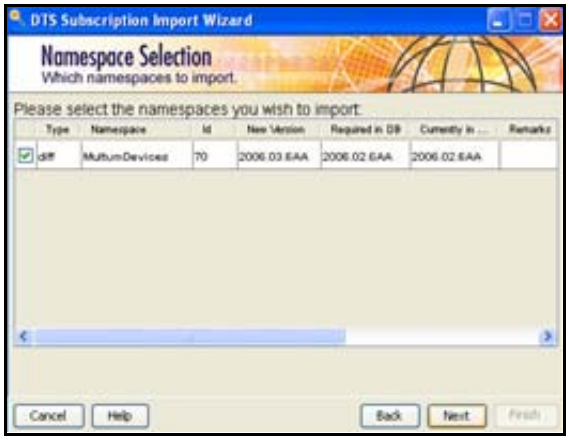


The default is to import all listed content and mapping update files from the .zip file. To deselect an individual file for the import, click the corresponding checkbox in the column adjacent to the **Type** column to remove the check mark. In the Namespace section of this illustrated window, the **DIFF** import will be prevented (an existing namespace version must exist in order to import updates to that namespace).

In the following illustration, the **Diff** import **(2006.04.06AA)** is disabled, as it requires a previous diff version **(2006.03.06AA)** in the target database (the current content version is **2006.02.6AA**).



In the next illustration, the **Diff** import **(2006.03.06AA)** is enabled, as it requires the previous diff version (2006.02.06AA) in the target database, which is present currently.



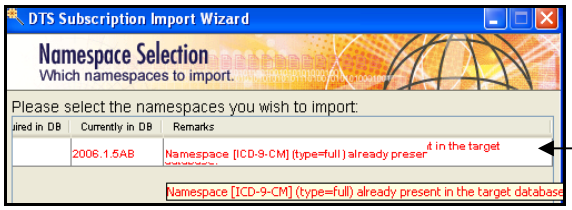
In the next illustration, the **DiffMap** import is enabled, as the versions required for its Source, Target, and Current namespaces are present in the database.

9. The subscription import update content for each namespace is accompanied by a configuration file (**kbcontent-import-diff.xml** for content, **kbmap-import-diff-[source_nsp_id]-[target_nsp_id].xml** for mapping) that includes information regarding the version of the current target namespace (if any) into which you are importing updated content. Using this information, the Import process will perform an integrity check to verify that the current (target) namespace version is compatible with the content update version you want to import.

   The import of update content for a specific namespace will be prevented if the Import process detects a version incompatibility between the updated content and the content in the target database currently.

An incompatibility is indicated in the *Namespace Selection* window if the values in the **Required in DB** and the **Currently in DB** fields are different. The checkbox for the namespace that is adjacent to the **Type** column is disabled, the text on that line is highlighted in red, and the **Remarks** column indicates the situation preventing the import.

In the following illustration, the subscription content version being imported exists already in the target namespace.



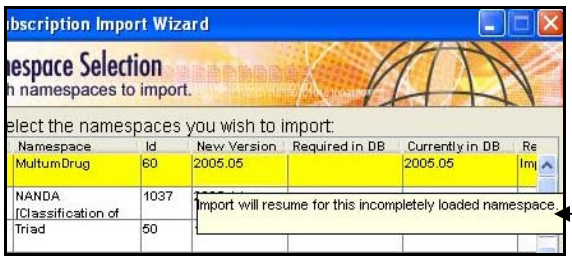
Namespace Already Present in Target Database

Note that the tooltip text box also indicates the incompatibility issue.

In the next illustration, a version mismatch is indicated (i.e., the namespace version under **New Version** differs from the version under **Required in DB**). This situation can occur if successive subscription updates are missed by a client.

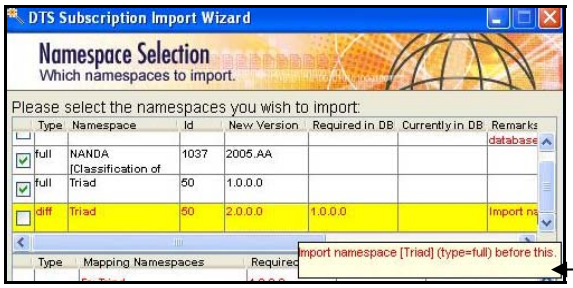
Namespace Version Mismatch

In the next illustration, an import that was interrupted earlier is being resumed (the namespace checkbox adjacent to the **Type** column is enabled in this case).



Resume Interrupted Import

The next illustration indicates that an update (i.e., **diff**) is being disabled because a **full** import of the namespace must be performed first.



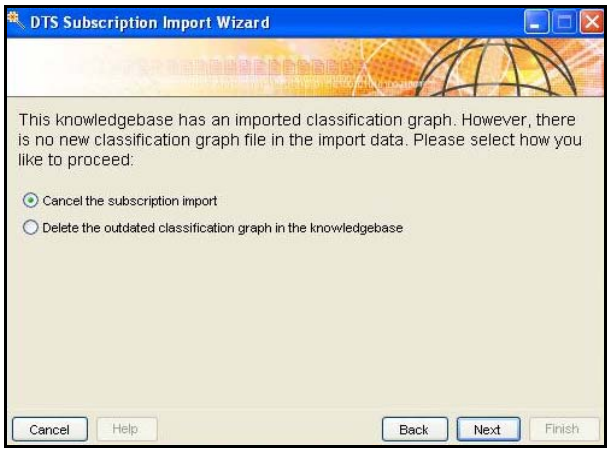
Full Import Required Prior to Update

The Import process will prevent import of subscription update content for a specific namespace until you reconcile the version incompatibility referenced for that namespace.

Import Classification Graph Only

As mentioned earlier, in order to run classification for an Ontylog Extension namespace, the linked subscription namespace must have a Classification Graph established for it. An updated version of the Ontylog subscription namespace's classification graph will be available for import with that namespace's content (**diff**) update.

If your current subscription namespace contains a classification graph, and you import the subscription update version without the updated classification graph, a warning message displays, along with options to continue the import (which deletes your current, outdated graph) or **Cancel** the subscription import.



The subscription update will terminate if you click **Cancel the subscription import**; at this point you can select the subscription update version that includes the updated version of the classification graph.

If you click **Delete the outdated version of the classification graph in the knowledgebase**, the import will proceed without the graph. The old version of the classifier graph is removed. You will **not** be able to classify the subscription's Extension namespaces until you import the graph with the version that matches the subscription version.

At a later time, you can import (only) the required classifier graph file. The **kbcontent-import-full.xml** configuration file (included with the import update content for each namespace) contains version information for the current target namespace into which you are importing updated content. The **kbcontent-import-full.xml** file is illustrated.

```
<?xml version="1.0" ?>

<!--
 ##########################################################################
 # This is configuration file used to import namespaces (sources)         #
 # provided in Common Data Format supplied by Apelon Inc.                 #
 ##########################################################################
-->

<!DOCTYPE DBConfig SYSTEM "http://apelon.com/dtd/util/db/dbconfig.dtd">
 <DBConfig>
        <!-- Content Type (full or diff) -->
        <property name="contentType" value="full" />

        <!-- Namespace(s) to import (Client KB) -->
        <namespace name="SNOMED CT" >
                <property name="add" value="true" />
        </namespace>

        <!-- New Version In (in CDF) -->
        <property name="newVersionIn" value="2005.07.5AB" />

        <!-- Version number for content (DTS_VERSION.ID in CDF) -->
        <property name="versionNumber" value="20050731" />

        <!-- Version Out (Client KB) -->
        <property name="versionOut" value="NOT_RETIRED" />
         <property name="graphOnly" value="true" />

        <!-- ~~~~ Advanced Setup ~~~~ -->
        <!--        importType=DATA/SQL  defualt value{DATA}}, fileExtension=.txt/.sql defualt{.txt} -->
        <!-- <property name="importType" value="SQL" /> -->
        <!-- <property name="fileExtension" value=".sql" /> -->

</DBConfig>
```

Add Property with Value "true"

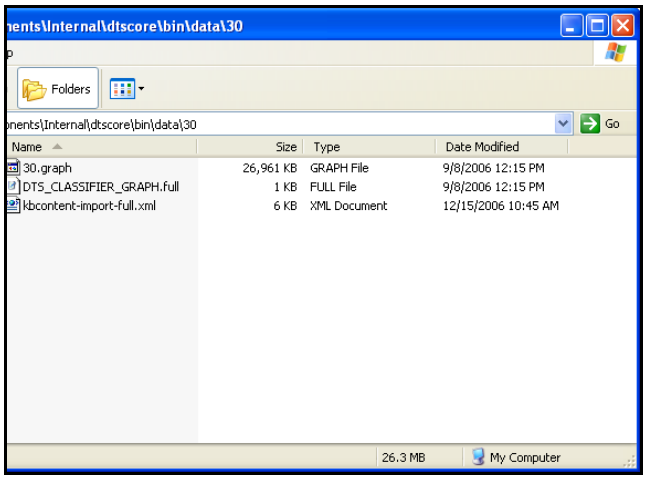
To perform the update for the graph only, you must add (if not present) the following property with the value "**true**," then save the **kbcontent-import-full.xml** file:

<property name="graphOnly" value="true" />

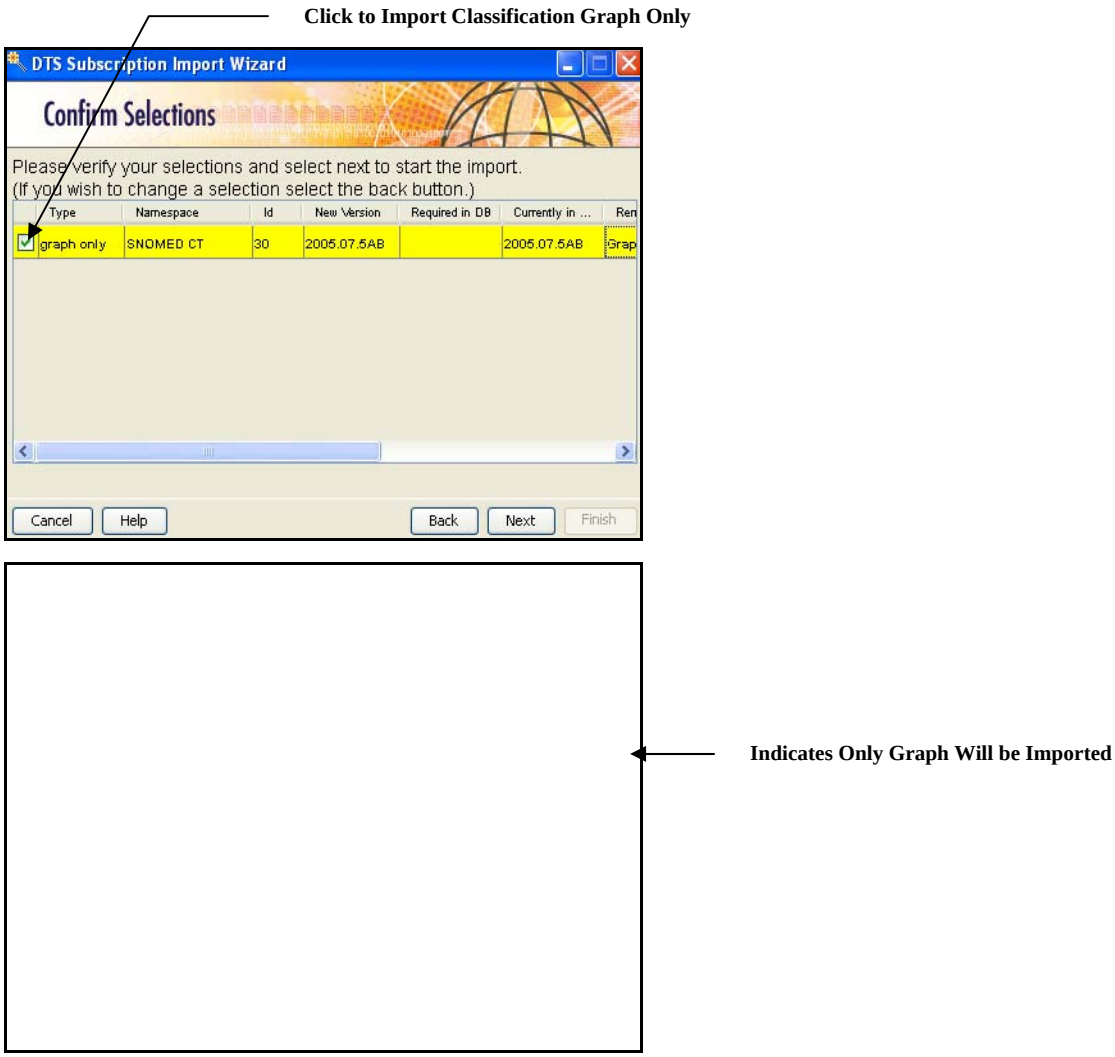
For graph only imports, content must contain 3 files only.

kbcontent-import-full.xml,
DTS_CLASSIFIER_GRAPH.full and
[SOURCE_ID].graph (as shown in figure below for SNOMED CT)

Note the files in the illustrated figure.

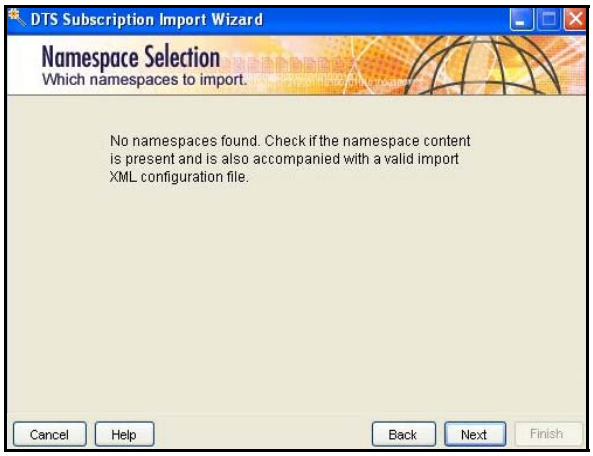


When you attempt to import the graph using the Import Wizard, the **Import Type** and **Remarks** reflect that you are importing the classification graph only.



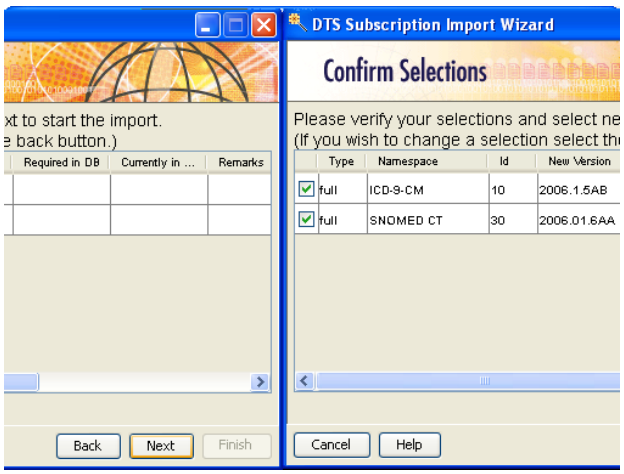
Import Subscription Update Content From DTS Versions Prior to 3.4

You cannot import subscription update content from DTS Version 3.3 (and older) unless there is a valid Import Integrity Verification configuration file (**kbcontent-import-diff.xml** for content, **kbmap-import-diff-[source_nsp_id]-[target_nsp_id].xml** for mapping) included with that older subscription content. If the configuration file is not included, the following window displays.



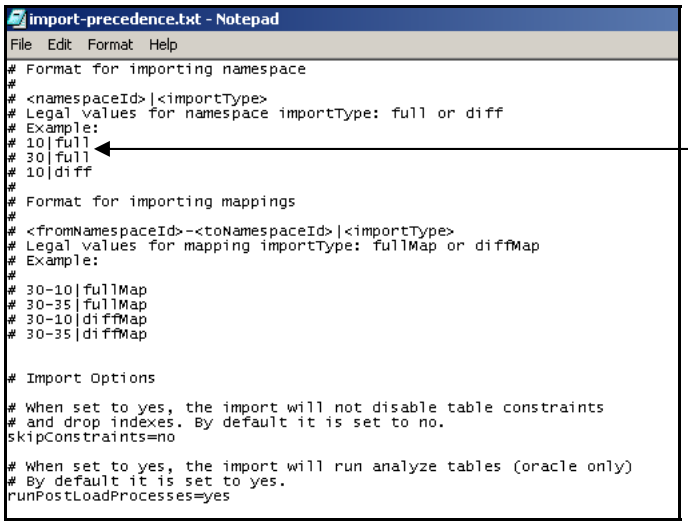
The Import Integrity Verification file must be generated and added to this older content before you can import that content into DTS. Refer to the *Generate Import Integrity Verification File* discussion in the *Knowledgebase Administrators Guide* for instructions on generating this configuration file, and including it with your older subscription content for import into DTS.

10. When you complete namespace and mapping file import selections, click **Next**. The *Confirm Selections* window displays for you to verify your selections.



If you need to modify your selections, click **Back**. When the *Namespace Selection* window redisplays you can modify your selections, as needed.

11. You have the option to define the order in which namespace updates will be imported. Open the **import-precedence.txt** file in ***DTSInstall* \bin**. The **import-precedence.txt** file is illustrated.

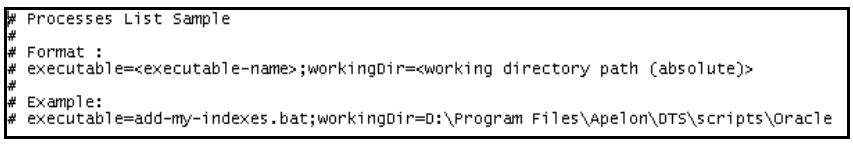


List of Namespace IDs

Modify the list of source namespace ID numbers to reflect the order in which you want to import the namespace update. Note that namespaces always are imported prior to their mappings to preserve database referential integrity.

Also included in the **import-precedence.txt** file are the **-pre** and **-post** switches. You use these switches to define tasks that can be run prior to the import process, or after the import completes (e.g., addition of indexes to archive tables **prior** to the import, deletion of archive table indexes **after** import completion).
You can use this feature to customize the import process with respect to performance improvements and tuning. Apelon subscriptions may include these options, depending on the size of the content updates.

Each switch must be accompanied by the task file name for the task to be run. The sample task file **pre-task-list-sample.txt** (in ***DTSInstall* \bin**) defines a task for the addition of indexes prior to the import run. The sample task file is illustrated.

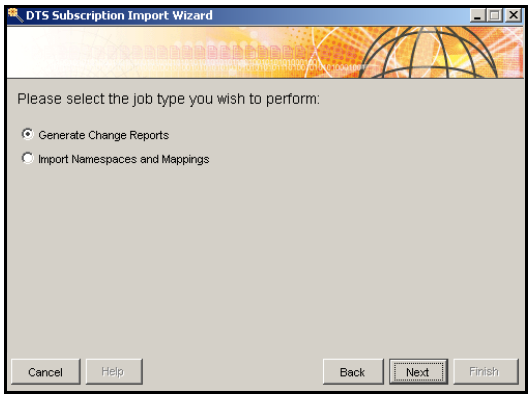


The required entry in the **import-precedence.txt** file is illustrated.

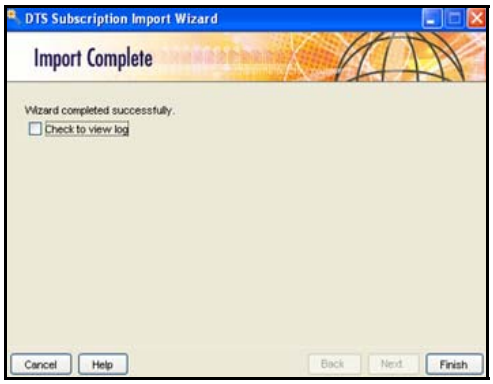
```
# Pre-processing tasks
# Provide a file name that has tasks listed which are required to be run
# before the import process
#
# pre=pre-task-list-sample.txt
# Post-processing tasks
# Provide a file name that has tasks listed which are required to be run
# after the import process
#
# post=
```

Task File Name

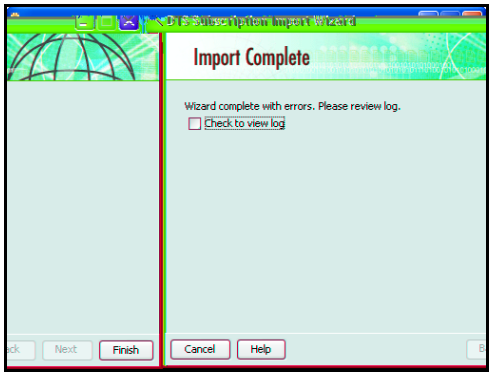
12. Click **Next**. The following window displays.



13. Click *Import Namespaces and Mappings* to begin the import process. A status window displays to indicate the progress of the import. A confirmation window displays when the import is completed. Note that it is recommended that you generate the *Content Change* reports for local namespace(s) with links to subscription content **before** you import the subscription updates.



If errors occurred during the import, click the *View Log* option to review them.



14. Click *Generate Change Reports* to generate two separate *Content Change* reports. The first *Content Change* report references concept changes, the second references term changes. As a result of each namespace subscription update, local namespace associations and/or properties will be broken for each linked subscription concept or term for which there has been a name, ID, or code change, as well as for each concept or term that has been marked in the subscription namespace as **retired** or **deleted**.

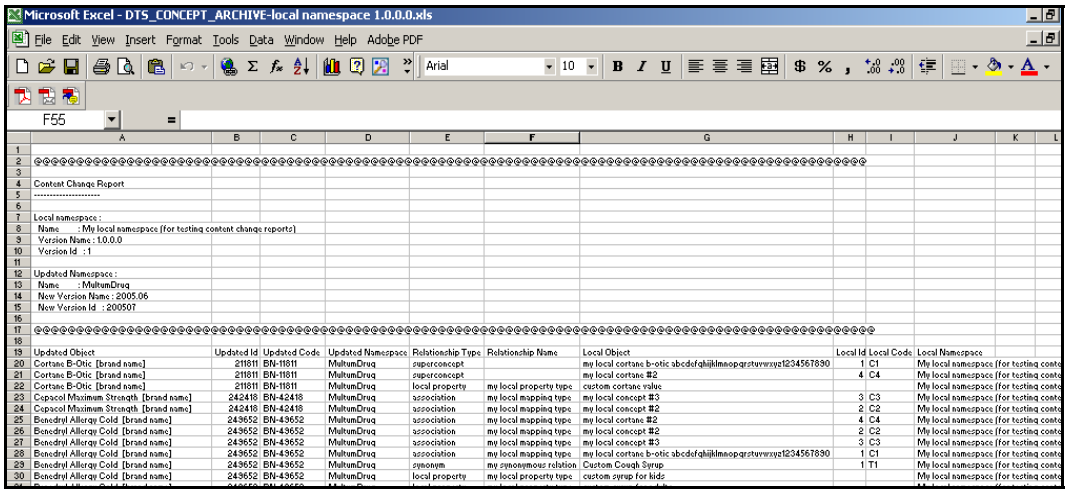
    The *Content Change* reports list all local properties, local associations, concept synonyms, and local role associations to subscription content that will be broken when you perform the import of the updated content. (Note that the reports do not reference qualifiers attached to listed local associations and properties.)

    It is recommended that you generate the *Content Change* reports for local namespace(s) with links to subscription content **before** you import the subscription updates. To insure that local namespace content is archived, you must first publish each local namespace for which you want to generate the Change Report prior to running the reports. This window displays if you do not first publish the local namespace (see *Publish Client's Local Namespace*, *Knowledgebase Administrators Guide*).

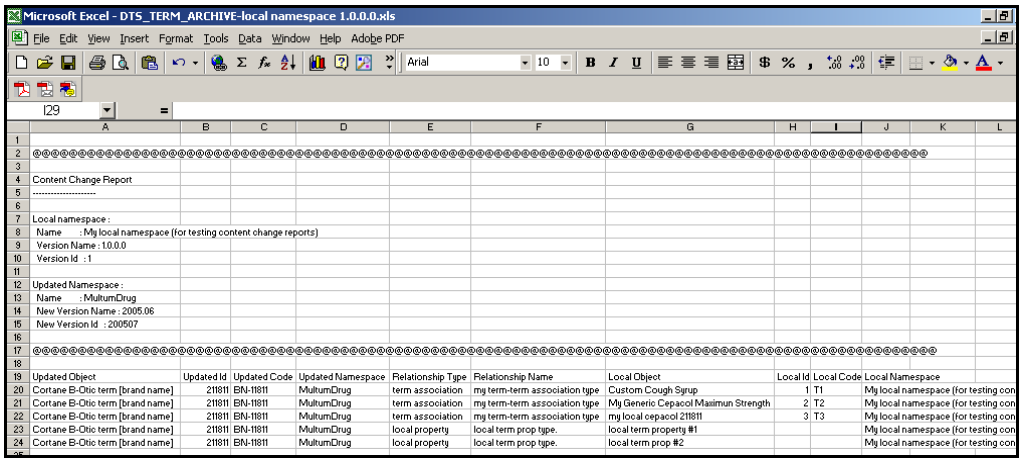
    The *Content Change* reports only **list** associations and properties that will be affected by update imports; report generation will not by itself repair broken local associations or properties. The **Local Content Repair** utility, which runs automatically on completion of the update import, restores (wherever possible) local property and association links that are broken as a result of the import. Properties and associations that could not be restored are listed in the report as well. Note the Local Content Repair discussion later in this section.
    As mentioned, **two** separate reports are generated when you select **Generate Change Reports**, each in tab-delimited format. By default, the reports are written to ***DTSInstall* \bin\data\report**.

The **Concept Changes** Report lists concept associations, roles, properties, and concept synonyms in the **local** namespace that will be affected by import of the updated subscription content. The following illustrated report (**DTS_CONCEPT_ARCHIVE-local namespace 1.0.0.0.rep**) reflects **concept** changes to published local namespace **1.0.0.0** that will result from the import of updated subscription content (this illustrated report was imported into an *Excel* spreadsheet).



The **Term Changes** Report lists term associations and properties in the local namespace that will be affected by the imported subscription updates.

The following illustrated report (**DTS_TERM_ARCHIVE-local namespace 1.0.0.0.rep**) reflects **term** changes to published local namespace **1.0.0.0** that will result from the import of updated subscription content (the illustrated report was imported into an *Excel* spreadsheet).



15. Click **Finish** to exit the Subscription Import Wizard.

If you cancel an import that is in progress by clicking the **Cancel** button, you can resume the import at the point at which it was stopped. Click *Import Namespaces and Mappings* to restart the import process at the point where it was interrupted.

As noted earlier, a **Local Content Repair** utility runs automatically after completion of the subscription update import. The utility attempts to restore local properties, roles, synonyms, and associations, as well as property and association qualifiers, that were lost (i.e., broken) as a result of the import. Local content can be broken due to deletion or retirement of a subscription concept or term, or a name change for a subscription concept or term.

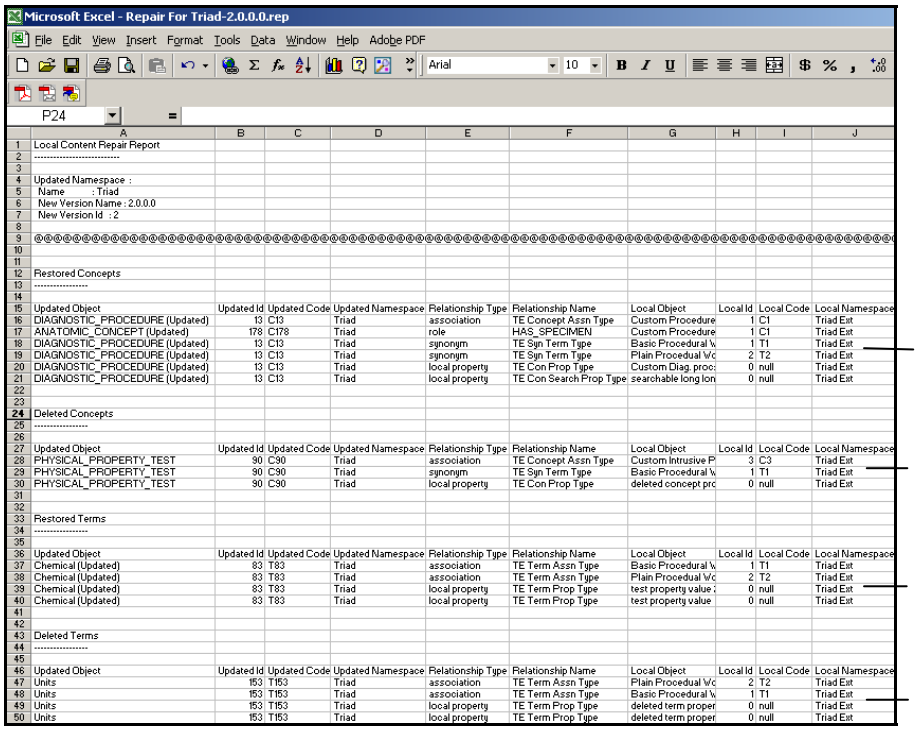
The **Local Content Repair Report** is generated automatically to reflect repaired content. The report is written to ***DTSInstall* \bin\data\report**, like the Content Change reports.

The following **Local Content Repair Report** illustration shows a report that was imported into an *Excel* spreadsheet. The **Restored Concepts** and **Restored Terms** sections list updated subscription concepts and/or terms restored by the utility. The **Deleted Concepts** and **Deleted Terms** sections list concepts and/or terms for which local content could not be restored.

In the example, the subscription namespace concept **DIAGNOSTIC_ PROCEDURE** was renamed (from **DIAGNOSTIC PROCEDURES**). The subscription concept has an association with a local namespace concept; this association is based on an association type that exists in the local namespace.

If the renamed subscription namespace concept **DIAGNOSTIC_PROCEDURE** is imported in the subscription update, the import of that concept name change breaks the existing (local) association to the local concept. The **Local Content Repair** utility restores that association between the renamed concept (now **DIAGNOSTIC_PROCEDURE**) and the local namespace concept.

Note that the **Local Content Repair Report** does not reference restored qualifiers that were attached to local associations and properties.



16. If you cancel an import that is in progress by clicking the **Cancel** button, you can resume the import at the point at which it was stopped. Click *Import Namespaces and Mappings* to restart the import process at the point where it was interrupted.

***** A modified version of the open source Java Wizard Framework was used in development of the DTS Subscription Import Wizard, which is included in this release. The Java Wizard framework and its use are covered by the GNU Lesser General Public License. A copy of the license is included in the **license.txt** file in the following installed DTS subdirectory:

***DTSInstall* \samples\wizard\license.txt**

You also can click the following link to view the license:

http://www.opensource.org/licenses/lgpl-license.php.

Perform "Diff" Import Using kbcontent-import.bat or kbcontent-import.sh

Follow this procedure to perform a **diff** subscription namespace import into your DTS schema by running **kbcontent-import.bat** file (for a DTS *Windows* installation), or, for a DTS Linux installation, by executing **kbcontent-import.sh** (in **bin/kb/content/import**). These scheduled updates are provided to you by Apelon based on your subscription agreements. Apelon **strongly** recommends that you perform a database backup prior to each subscription update import.

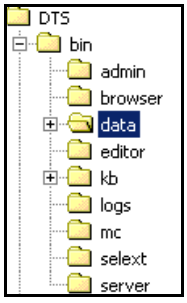
DTS allows you to create new content for each Ontylog subscription namespace in a linked **Extension** namespace. Each Extension namespace extends the content in a specific linked subscription namespace. Ontylog namespace concepts are organized into a hierarchy by a process called **classification**, a process you can perform using the DTS Editor.

In order to run classification for an Extension namespace, the linked subscription namespace must have a Classification Graph established for it. Refer to the *Import Ontylog Subscription Updates With "Classification" Graph* discussions in the *Ontylog Extension Namespaces and Extension Namespace Classification in DTS* document for more information.

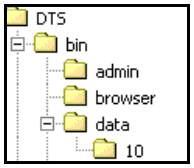
> Note that using the **Knowledgebase Publishing** utility provided with Apelon DTS, you also can publish your local namespace content on a regular basis, and forward your own updated content to **your** subscribers. Refer to the *Publish Client's Local Namespace* discussions in the *Knowledgebase Administrators Guide* for procedures.

Follow this procedure to import each subscription update into your knowledgebase.

1. Unzip the **SOURCE-VERSION-DIFF.zip** import data file provided by Apelon into the ***DTSInstall*\bin** directory. A **data** subdirectory is created in ***DTSInstall*\bin**.



   Under the new **data** directory, a separate subdirectory is created for each namespace source (i.e., namespace) in your knowledgebase. Note the illustration.



2. At this point you must edit the **kbcontent-import.bat** file in ***DTSInstall*** **\bin\kb\content\import** for the import. (For a Linux installation, refer to the discussion on executing **kbcontent-import.sh** discussion later in this section.)

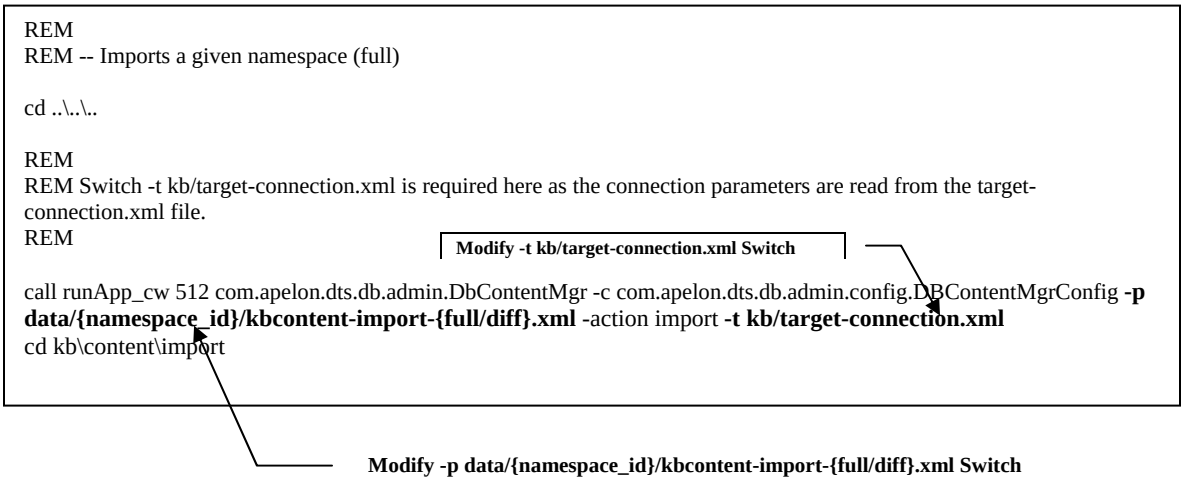
   The subscription update import content for each namespace is accompanied by a configuration file (**kbcontent-import-diff.xml** for content, **kbmap-import-diff-[source_nsp_id]-[target_nsp_id].xml** for mapping). Each file includes information regarding the version of the current target namespace (if any) into which you are importing updated content. The configuration file for each namespace to be imported will be in ***DTSInstall*** **\bin\data\[namespace_id].**

   Using this information, the Import process will perform an integrity check to verify that the current (target) namespace version is compatible with the version you want to import. The import of a specific namespace will be prevented if the process detects a version incompatibility between import namespace content and the content in the target database.

   Note the highlighted switches in the illustrated **kbcontent-import.bat** file.

```
@echo off
REM
REM Process Arguments : -p <properties-file.xml> -c <configuration full-class name>
REM

REM
REM Process arguments overriding the properties file configuration variables:
REM (i.e., connection params from a separate file overrides one in the property file):
REM Usage: -c <system configuration class name> which is mandatory
REM         -action [export, import, export_mappings, import_mappings, content_change_report, publish_namespace]
REM                 export -> exporting content from DTS
REM                 import -> importing content into DTS
REM                 export_mappings -> exporting inter-source mappings (associations) from DTS knowledgebase
REM                 import_mappings -> importing inter-source mappings (associations) into DTS knowledgebase
REM                 content_change_report -> to generate reports showing concept associations affected by
subscription content update
REM                 publish_namespace -> publish a local namespace
REM         -p <user property file name>
REM                 XML file having the properties needed to run this process (supplied with namespace content).
REM         -skipConstraints (yes/no)
REM                 If yes, the process does not disable/enable constraints and drop/add indexes while updating DTS
knowledgebase
REM         -t <target connection config file>
REM                 file with the target connection params.
REM                 The application loads connection params from this file, and NOT from kbcontent-import-
{full/diff}.xml
REM         -d <data extract directory>
REM                 path to folder where subscription content is to be read from/written to.
REM         -runPostLoadProcesses <post load processing (yes/no) {optional}, default=yes>
REM          [e.g. analyze tables in case of Oracle]
REM       -pre  <pre processes file name>
REM       -post <post processes file name>
```



Edit the **-p data/{namespace_id}/kbcontent-import-{full/diff}.xml** switch to reference the import configuration file in ***DTSInstall* \bin\data\ [namespace_id]** that accompanied the updated namespace content; include a switch for each updated source namespace.

You can edit the **kbcontent-import.bat** file (or **kbcontent-import.sh,** for Linux) to reference a custom connection configuration file that will not be overwritten during the subscription update process. The template connection file **target-connection.xml** allows you to define a custom connection based on the **target** connection parameters for content import. To establish a target connection in the **target-connection.xml** file, the value of the **direction** property should be **target**.

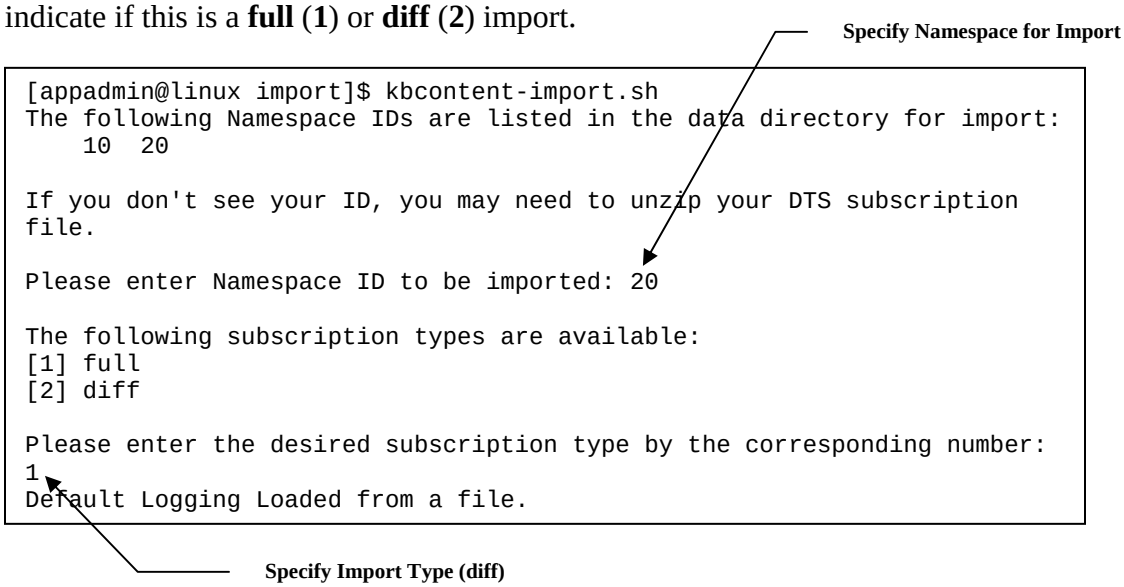
Note the illustration.

<property name="direction" value="**target**" />

You can create multiple versions of the **target-connection.xml** file, each with its own specific name, and each reflecting a specific connection from, or to, a data source.

Modify the **-t kb/target-connection.xml** switch to reference the appropriate target-connection.xml file. Apelon recommends that you rename each modified **kbcontent-import.bat** batch file (or **kbcontent-import.sh,** for Linux) to reflect its purpose (e.g., **kbcontent-update-import-test.bat**, **kbcontent-update-import-production.bat**, etc.).

3. Run **kbcontent-import.bat** (or **kbcontent-import.sh,** for Linux) to import updated subscription content into your knowledgebase. Updated (diff) content from each namespace source referenced in the **kbcontent-import.bat** file (or **kbcontent-import.sh,** for Linux) is imported from its corresponding subscription namespace subdirectory under ***DTSInstall*** **\bin\data**.
When you execute **kbcontent-import.sh** (**bin/kb/content/import**) for Linux, you are prompted to specify the IDs for the namespaces you want to import, and indicate if this is a **full** (**1**) or **diff** (**2**) import.



If the Import process detects a version incompatibility that would cause the individual namespace import to fail, the import of that namespace is prevented. You must reconcile the version incompatibility for the namespace that failed the import, then attempt the import again.

4. The **Content Change Reports** (two separate reports are generated) list the content of a local namespace that might be affected when you perform an import of updated subscription content (e.g., local associations to subscription content, local properties for subscription content). Local namespace associations and/or properties will be broken for concepts that are marked as **retired**/**deleted** in the standard namespace subscription update.

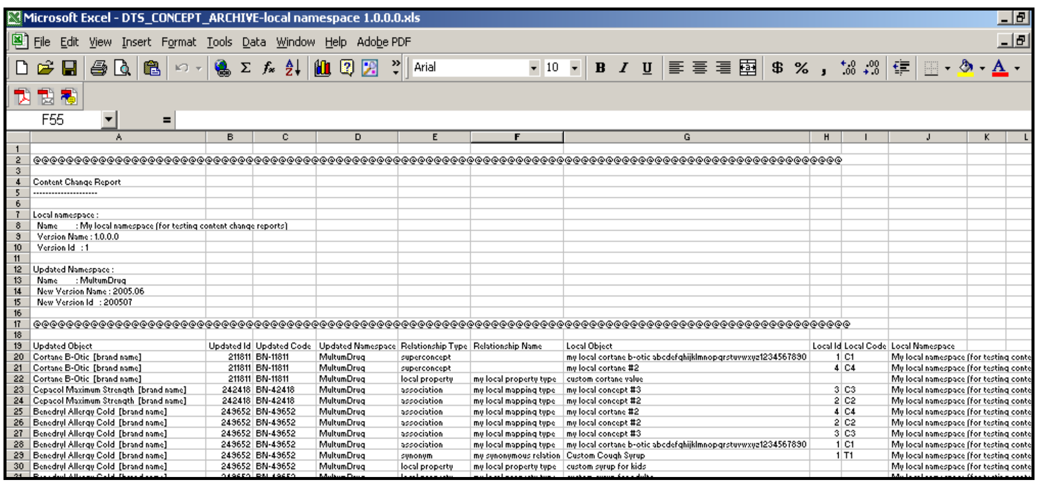
   You run the Content Change Reports by running the file **content-change-report.bat** in ***DTSInstall*** **\bin\kb\content\report** (for a Linux installation, execute **content-change-report.sh** in **bin/kb/content/report**). Modify the **kbcontent-report.xml** configuration file (***DTSInstall*** **\bin\kb\content\report**) to reflect the appropriate database connections, then run **content-change-report.bat**.

   It is recommended that you generate Change reports for the local namespace(s) that have links to subscription content **before** you import the subscription updates.

The *Content Change* reports only **list** associations and properties that will be affected by update imports; report generation will not by itself repair broken local associations or properties. The **Local Content Repair** utility, which runs automatically on completion of the update import, restores (wherever possible) local property and association links that are broken as a result of the update import.
Properties and associations that could not be restored are listed in the report as well. Note the Local Content Repair discussion later in this section.

As mentioned, **two** separate reports are generated, each in tab-delimited format; by default, the reports are written to ***DTSInstall* \bin\data\report**. The **Concept Changes** report lists concept associations, concept properties, and concept synonyms in the **local** namespace that will be affected by import of the updated **subscription** content.

The following sample report reflects local namespace **concept** changes that will result from the import. The report **DTS_CONCEPT_ ARCHIVE-local namespace 1.0.0.0.rep** was imported into an *Excel* spreadsheet, and is illustrated.


Microsoft Excel - DTS_CONCEPT_ARCHIVE-local namespace 1.0.0.0.xls

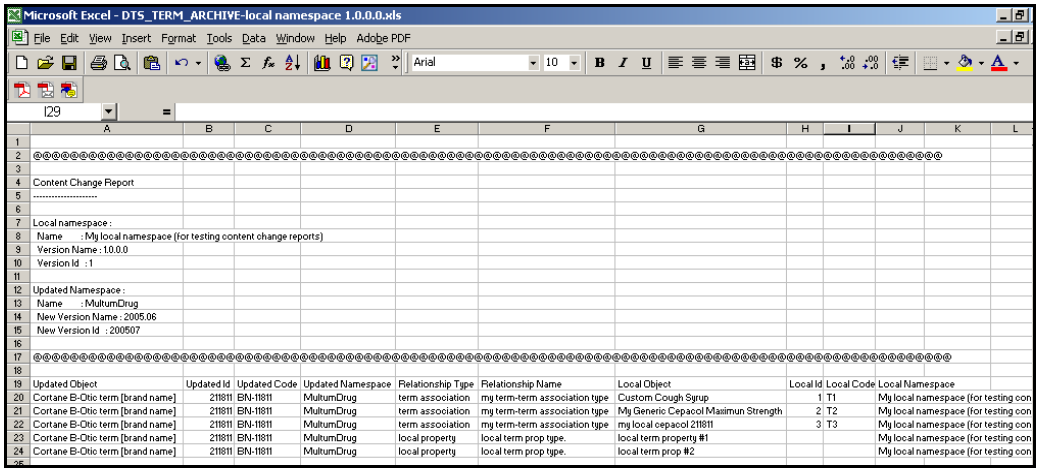
Content Change Report

Local namespace :
Name : My local namespace (for testing content change reports)
Version Name : 1.0.0.0
Version Id : 1

Updated Namespace :
Name : MultumDrug
New Version Name : 2005.06
New Version Id : 200507

Updated Object	Updated Id	Updated Code	Updated Namespace	Relationship Type	Relationship Name	Local Object	Local Id	Local Code	Local Namespace
Cortane B-Otic [brand name]	211811	BN-11811	MultumDrug	superconcept		my local cortane b-otic abcdefghijklmnopqrstuvwxyz1234567890	1	C1	My local namespace (for testing conte
Cortane B-Otic [brand name]	211811	BN-11811	MultumDrug	superconcept		my local cortane #2	4	C4	My local namespace (for testing conte
Cortane B-Otic [brand name]	211811	BN-11811	MultumDrug	local property	my local property type	custom cortane value			My local namespace (for testing conte
Cepacol Maximum Strength [brand name]	242418	BN-42418	MultumDrug	association	my local mapping type	my local concept #3	3	C3	My local namespace (for testing conte
Cepacol Maximum Strength [brand name]	242418	BN-42418	MultumDrug	association	my local mapping type	my local concept #2	2	C2	My local namespace (for testing conte
Benedryl Allergy Cold [brand name]	249652	BN-49652	MultumDrug	association	my local mapping type	my local cortane #2	4	C4	My local namespace (for testing conte
Benedryl Allergy Cold [brand name]	249652	BN-49652	MultumDrug	association	my local mapping type	my local concept #2	2	C2	My local namespace (for testing conte
Benedryl Allergy Cold [brand name]	249652	BN-49652	MultumDrug	association	my local mapping type	my local concept #3	3	C3	My local namespace (for testing conte
Benedryl Allergy Cold [brand name]	249652	BN-49652	MultumDrug	association	my local mapping type	my local cortane b-otic abcdefghijklmnopqrstuvwxyz1234567890	1	C1	My local namespace (for testing conte
Benedryl Allergy Cold [brand name]	249652	BN-49652	MultumDrug	synonym	my synonymous relation	Custom Cough Syrup	1	T1	My local namespace (for testing conte
Benedryl Allergy Cold [brand name]	249652	BN-49652	MultumDrug	local property	my local property type	custom syrup for kids			My local namespace (for testing conte

The **Term Changes** report lists term associations and term properties in the local namespace that will be affected by the updated subscription content to be imported. The report **DTS_TERM_ ARCHIVE-local namespace 1.0.0.0.rep** was imported into an *Excel* spreadsheet, and is illustrated.



Microsoft Excel - DTS_TERM_ARCHIVE-local namespace 1.0.0.0.xls

Content Change Report

Local namespace :
Name : My local namespace (for testing content change reports)
Version Name : 1.0.0.0
Version Id : 1

Updated Namespace :
Name : MultumDrug
New Version Name : 2005.06
New Version Id : 200507

Updated Object	Updated Id	Updated Code	Updated Namespace	Relationship Type	Relationship Name	Local Object	Local Id	Local Code	Local Namespace
Cortane B-Otic term [brand name]	211811	BN-11811	MultumDrug	term association	my term-term association type	Custom Cough Syrup	1	T1	My local namespace (for testing con
Cortane B-Otic term [brand name]	211811	BN-11811	MultumDrug	term association	my term-term association type	My Generic Cepacol Maximun Strength	2	T2	My local namespace (for testing con
Cortane B-Otic term [brand name]	211811	BN-11811	MultumDrug	term association	my term-term association type	my local cepacol 211811	3	T3	My local namespace (for testing con
Cortane B-Otic term [brand name]	211811	BN-11811	MultumDrug	local property	local term prop type.	local term property #1			My local namespace (for testing con
Cortane B-Otic term [brand name]	211811	BN-11811	MultumDrug	local property	local term prop type.	local term prop #2			My local namespace (for testing con

5. Run the **kbcontent-import.bat** file to start the subscription update import (for a Linux installation, execute **content-change-report.sh** start the update import).

   As noted earlier, a **Local Content Repair** utility runs automatically after completion of the update import. The utility attempts to restore local properties, roles, synonyms, and associations, as well as property and association qualifiers, that were lost (i.e., broken) as a result of the import. Local content can be broken due to deletion or retirement of a subscription concept or term, or a name change for a subscription concept or term.

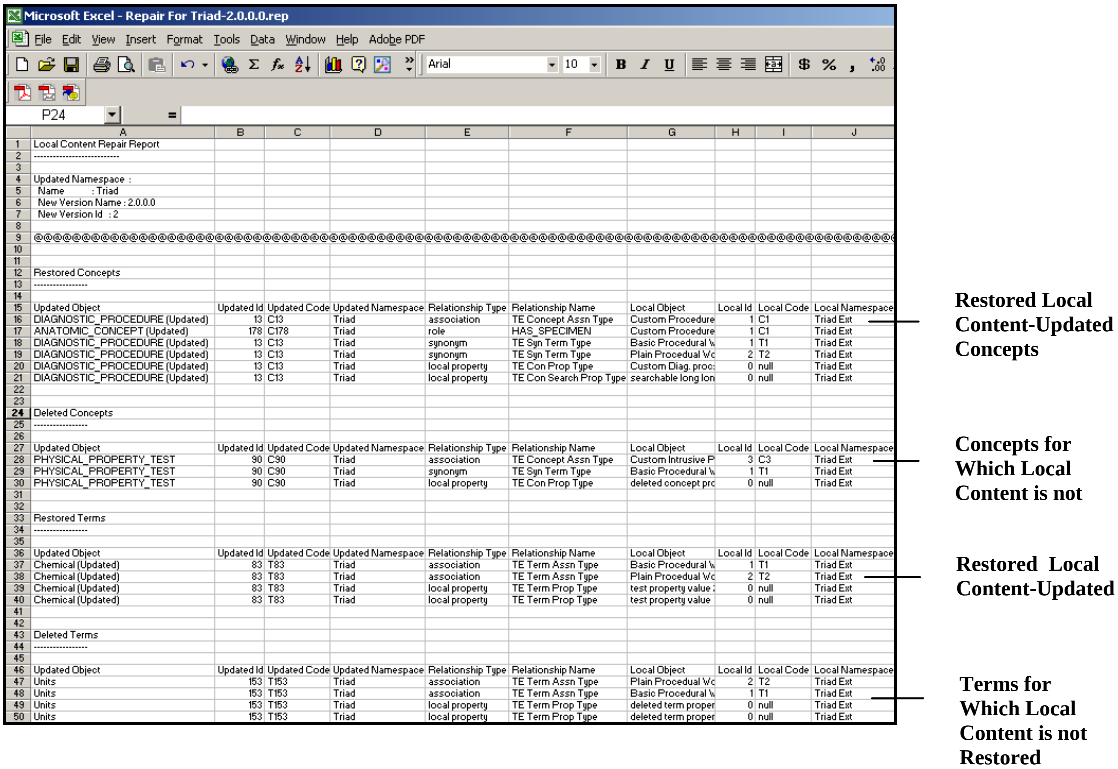
   The **Local Content Repair Report** is generated automatically to reflect repaired content. The report is written to ***DTSInstall* \bin\data\report**, like the Content Change reports.

   The following **Local Content Repair Report** illustration shows a report that was imported into an *Excel* spreadsheet. The **Restored Concepts** and **Restored Terms** sections list updated subscription concepts and/or terms restored by the utility. The **Deleted Concepts** and **Deleted Terms** sections list concepts and/or terms for which local content could not be restored.

   In the example, the subscription namespace concept **DIAGNOSTIC_ PROCEDURE** was renamed (from **DIAGNOSTIC PROCEDURES**). The subscription concept has an association with a local namespace concept; this association is based on an association type that exists in the local namespace.

   If the renamed subscription namespace concept **DIAGNOSTIC_PROCEDURE** is imported in the subscription update, the import of that concept name change breaks the existing (local) association to the local concept. The **Local Content Repair** utility restores that association between the renamed concept (now **DIAGNOSTIC_PROCEDURE**) and the local namespace concept.

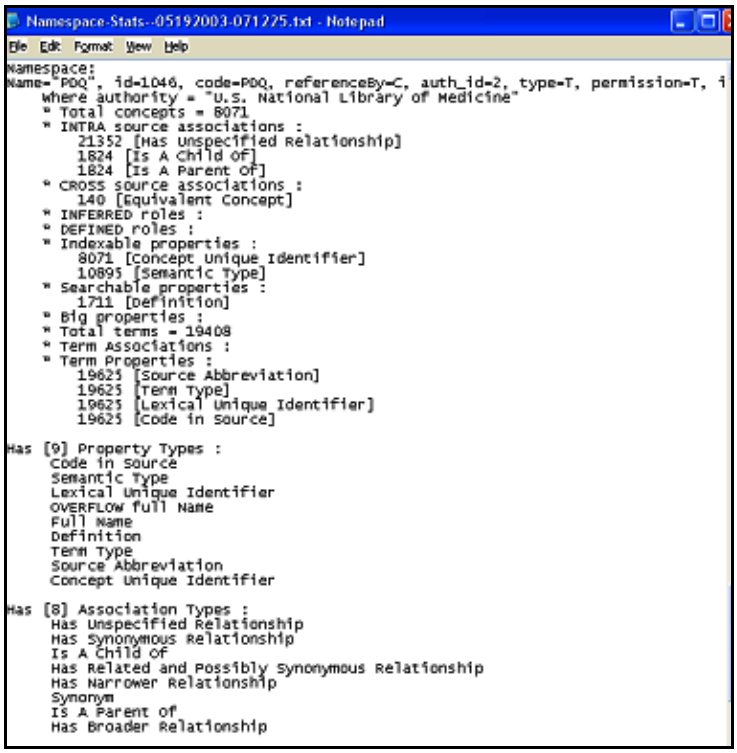
Note that the **Local Content Repair Report** does not reference restored qualifiers that were attached to local associations and properties.



	A	B	C	D	E	F	G	H	I	J
1	Local Content Repair Report									
2	--------------------------									
3										
4	Updated Namespace :									
5	Name : Triad									
6	New Version Name : 2.0.0.0									
7	New Version Id : 2									
8										
9	@@@									
10										
11										
12	Restored Concepts									
13	-----------------									
14										
15	Updated Object	Updated Id	Updated Code	Updated Namespace	Relationship Type	Relationship Name	Local Object	Local Id	Local Code	Local Namespace
16	DIAGNOSTIC_PROCEDURE (Updated)	13	C13	Triad	association	TE Concept Assn Type	Custom Procedure	1	C1	Triad Ext
17	ANATOMIC_CONCEPT (Updated)	178	C178	Triad	role	HAS_SPECIMEN	Custom Procedure	1	C1	Triad Ext
18	DIAGNOSTIC_PROCEDURE (Updated)	13	C13	Triad	synonym	TE Syn Term Type	Basic Procedural W	1	T1	Triad Ext
19	DIAGNOSTIC_PROCEDURE (Updated)	13	C13	Triad	synonym	TE Syn Term Type	Plain Procedual Wo	2	T2	Triad Ext
20	DIAGNOSTIC_PROCEDURE (Updated)	13	C13	Triad	local property	TE Con Prop Type	Custom Diag. proc:	0	null	Triad Ext
21	DIAGNOSTIC_PROCEDURE (Updated)	13	C13	Triad	local property	TE Con Search Prop Type	searchable long lon	0	null	Triad Ext
22										
23										
24	Deleted Concepts									
25	-----------------									
26										
27	Updated Object	Updated Id	Updated Code	Updated Namespace	Relationship Type	Relationship Name	Local Object	Local Id	Local Code	Local Namespace
28	PHYSICAL_PROPERTY_TEST	90	C90	Triad	association	TE Concept Assn Type	Custom Intrusive P	3	C3	Triad Ext
29	PHYSICAL_PROPERTY_TEST	90	C90	Triad	synonym	TE Syn Term Type	Basic Procedural W	1	T1	Triad Ext
30	PHYSICAL_PROPERTY_TEST	90	C90	Triad	local property	TE Con Prop Type	deleted concept pro	0	null	Triad Ext
31										
32										
33	Restored Terms									
34	-----------------									
35										
36	Updated Object	Updated Id	Updated Code	Updated Namespace	Relationship Type	Relationship Name	Local Object	Local Id	Local Code	Local Namespace
37	Chemical (Updated)	83	T83	Triad	association	TE Term Assn Type	Basic Procedural W	1	T1	Triad Ext
38	Chemical (Updated)	83	T83	Triad	association	TE Term Assn Type	Plain Procedual Wo	2	T2	Triad Ext
39	Chemical (Updated)	83	T83	Triad	local property	TE Term Prop Type	test property value :	0	null	Triad Ext
40	Chemical (Updated)	83	T83	Triad	local property	TE Term Prop Type	test property value	0	null	Triad Ext
41										
42										
43	Deleted Terms									
44	-----------------									
45										
46	Updated Object	Updated Id	Updated Code	Updated Namespace	Relationship Type	Relationship Name	Local Object	Local Id	Local Code	Local Namespace
47	Units	153	T153	Triad	association	TE Term Assn Type	Plain Procedual Wo	2	T2	Triad Ext
48	Units	153	T153	Triad	association	TE Term Assn Type	Basic Procedural W	1	T1	Triad Ext
49	Units	153	T153	Triad	local property	TE Term Prop Type	deleted term proper	0	null	Triad Ext
50	Units	153	T153	Triad	local property	TE Term Prop Type	deleted term proper	0	null	Triad Ext

6. Run **namespaceStats.bat** (***DTSInstall*** **\bin\kb\load**) to generate the **Namespace Stats Report** (in .txt format) and write it to the ***DTSInstall*** **\bin\kb\content\import** directory. For a Linux installation, execute **namespaceStats.sh** in **bin/kb/load** to generate the **Namespace Stats Report**.

   The report lists statistics by namespace(s) for each subscription update import, information you can use to validate the subscription import. Note the illustration.



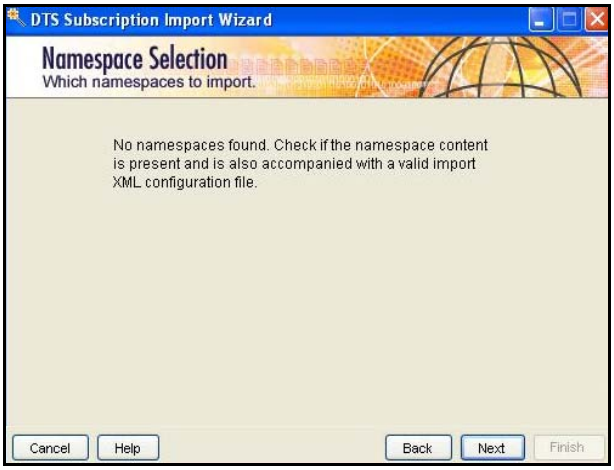
After you publish your local namespace content, package it, then forward it to your subscription client (subscriber) the subscriber must, in turn, use the **Client Import** utility to import your local content.

Note: The updated subscription import content for each namespace is accompanied by a configuration file (**kbcontent-import-diff.xml** for content, **kbmap-import-diff-[source_nsp_id]-[target_nsp_id].xml** for mapping) that includes information regarding the version of the current target namespace (if any) into which you are importing updated content. Using this information, the Import process performs an integrity check to verify that the current (target) namespace is compatible with the update content you are importing.

If the Import process detects a version incompatibility that would cause the individual namespace import to fail, the import of that namespace is prevented. You must reconcile the version incompatibility for the namespace that failed the import, then attempt the import again.

Import Updated Content From DTS Versions Prior to 3.4

You cannot import updated content from DTS Version 3.3 (and older) unless there is a valid Import Integrity Verification configuration file (**kbcontent-import-diff.xml** for content, **kbmap-import-diff-[source_nsp_id]-[target_nsp_id].xml** for mapping) included with that older subscription content. If the configuration file is not included, the following window displays.



This Import Integrity Verification file must be generated and added to this older content before you can import that content into DTS. Refer to the *Generate Import Integrity Verification File* discussion in the *Knowledgebase Administrators Guide* for instructions on generating this file, and for including it with your older subscription content for import into DTS.

Back to Top